



ESCUELA TÉCNICA SUPERIOR DE INGENIEROS INDUSTRIALES Y DE TELECOMUNICACIÓN

Titulación:

INGENIERO DE TELECOMUNICACIÓN

Título del proyecto:

ANÁLISIS DE TÉCNICAS Y SISTEMAS DE LOCALIZACIÓN INDOOR EN REDES INALÁMBRICAS

César Guillén Salvador

Dr. Francisco Javier Falcone Lanas

Pamplona, 25 de Junio de 2010



UNIVERSIDAD PÚBLICA DE NAVARRA

E.T.S.I.I. y de Telecomunicación

Campus de Arrosadía s/n

31006-Pamplona Navarra-Spain

Tfno: 34-48-169000 Fax: 34-48-169169



Acta de Proyecto Final de Carrera

Título:

***“ANÁLISIS DE TÉCNICAS Y SISTEMAS DE
LOCALIZACIÓN INDOOR EN REDES
INALÁMBRICAS”***

Alumno:

César Guillén Salvador

Tutor del Proyecto:

Dr. Francisco Javier Falcone Lanas

Reunido el tribunal, compuesto por los siguientes miembros,

Presidente,

Secretario,

Vocal 1º,

Fdo:.....

Fdo:.....

Fdo:.....

acuerda otorgar la calificación de

Pamplona,de.....de 20.....

Agradecimientos

Estas líneas pretenden resumir 7 años de estudio muy intensos. Cuando todo el mundo decía lo difícil que iba a ser la carrera, la verdad es que no se equivocaban. He vivido buenos y malos momentos pero el resumen es simple... merece la pena!

Durante estos años he hecho muy buenos amigos y de hecho, sin la ayuda de muchos de ellos este momento no creo que hubiera llegado. La verdad es que tras estos años no puedo hablar de compañeros de clase sino de verdaderos amigos ya que al final nos hemos convertido en una piña y eso se nota.

Todos recordamos los buenos momentos de los viajes, cenas o las casas rurales pero los momentos vividos en la biblioteca, en el txoko de la escuela durante esos largos descansos o en nuestra segunda casa los últimos años, el laboratorio de TDS, han sido nuestro día a día y también los recordaré con una sonrisa.

Los profesores también son partícipes de esto por su paciencia y dedicación conmigo. Por ello, me llevo buenos momentos y recuerdos de ellos.

A Eli como no. Una de las razones que me alegra de haber elegido ser Teleco. Una persona que ha pasado de ser mi compañera, para ser mi amiga y finalmente se ha convertido en una de las personas más importantes de mi vida. Muchas gracias por estar ahí a mi lado ya que sin ti esto no hubiera sido igual. Gracias por ser como eres! Y gracias a tu familia por convertirme en uno mas!

No puedo dejar de mencionar a los grandes apoyos que he tenido fuera de la universidad que han conseguido que siguiera para adelante. Durante los primeros años no voy a negar que pensara en abandonar este largo camino, pero con la ayuda de mis amigos de siempre y mi familia cambié de idea.

Mis padres de hecho se merecen un monumento. A pesar de no entender muy bien cómo funcionaba eso de los créditos o no entender quien era ese tal Fourier que me volvía loco, ahí han estado para aguantar mis quejas y para animarme en los momentos difíciles. Por ello, ahora que es un momento feliz, también quiero que lo pasen conmigo porque se lo merecen. Gracias!

A mi cuadrilla, por aguantarme cada finde que iba al pueblo. Por escucharme e intentar comprenderme cuando les decía que había estudiado durante 1 mes sólo una asignatura llamada antenas, para sacar un 3. Por todo ello, una parte de este éxito para mí se lo debo a ellos.

A los que me han ayudado a realizar este proyecto final de carrera. Primero sin duda a Francisco por acogerme cuando me encontraba en un momento difícil del proyecto y por la dedicación que ha empleado en mí. Espero haber cumplido tus expectativas. Y segundo, a mis amigos de clase Fermín y Victor. A Fermín porque me ha ayudado a entender ese algoritmo tan raro que hizo pero a la vez tan útil para mi proyecto y a Victor por su ayuda con los escenarios y por aguantar mis interminables preguntas. Gracias a los tres!!!

A IECISA y en especial al departamento de Ingeniería. Por haber sido mi casa durante estos últimos meses. Ahí comenzó el proyecto y ahí ha finalizado. Muchas gracias por todo!

Por último a todo el mundo que no aparece de manera explícita pero me ha ayudado durante todo este tiempo.

MUCHAS GRACIAS!!!!

TABLA DE CONTENIDO

INDICE DE FIGURAS Y TABLAS	5
INTRODUCCIÓN	5
1.1. OBJETIVOS	11
1.2. ESTRUCTURA DEL PROYECTO	11
ESTADO DEL ARTE	12
2.1. CANAL RADIO	14
2.1.1. Desvanecimientos a gran escala	14
2.1.1.1. Atenuación	15
2.1.1.2. Refracción	15
2.1.1.3. Reflexión	16
2.1.1.4. Difracción	16
2.1.1.5. Dispersión	17
2.1.2. Desvanecimientos a pequeña escala	18
2.1.2.1. Propagación multitrayecto	18
2.1.2.2. Efecto Doppler	20
2.2. TECNOLOGÍAS PARA LA LOCALIZACIÓN	23
2.2.1. GPS	24
2.2.2. Galileo	25
2.2.3. 802.11	26
2.2.4. RFID	30
2.2.5. Localización utilizando la telefonía móvil	31
2.2.6. Localización por Infrarrojos	31
2.2.7. Bluetooth	32
2.2.8. UWB (Ultra Wideband)	33
2.2.9. Zig-Bee	34
2.2.10. Comparativa	35
2.3. TÉCNICAS DE LOCALIZACIÓN	37

2.3.1.	TOA.....	37
2.3.2.	AoA	39
2.3.3.	RSS.....	39
2.3.4.	Tabla comparativa	40
TÉCNICAS Y SISTEMAS DE LOCALIZACIÓN		41
3.1.	TIME DIFFERENCE OF ARRIVAL	42
3.1.1.	Descripción matemática	44
3.1.2.	Algoritmos de resolución de ecuaciones no lineales hiperbólicas	48
3.1.2.1.	Método de las series de Taylor	48
3.1.2.2.	Método de Fang	50
3.1.2.3.	Método de interpolación esférica.....	54
3.1.2.4.	Método de Chan.....	57
3.1.2.5.	Método de Friedlander	62
3.1.2.6.	Método de Levenberg - Marquardt	64
3.1.2.7.	Método Divide-and-Conquer	66
3.1.2.8.	Método de Steepest Descent	67
3.1.3.	Tabla comparativa	69
3.2.	RECEIVED SIGNAL STRENGTH	71
3.2.1.	Métodos de estimación de localización basados en RSS	71
3.2.1.1.	Triangulación	71
3.2.1.2.	Reconocimiento de patrones	72
	• Los métodos KNN.....	73
	• Métodos bayesianos	74
	• Métodos basados en redes neuronales.....	76
3.3.	RESULTADOS DE LAS SIMULACIONES.....	77
3.3.1.	Estudio preliminar	78
3.3.2.	Estudio avanzado.....	81
3.3.2.1.	Descripción de los escenarios	83
3.3.2.2.	Resultados obtenidos	86
3.3.2.2.1.	TDOA	86
3.3.2.2.2.	RSS	95

PROYECTO REAL	97
4.1. PROYECTO REAL IECISA.....	98
 CONCLUSIONES	 104
5.1. CONCLUSIONES.....	105
5.2. LÍNEAS FUTURAS.....	107
 ANEXOS.....	 108
6.1. BIBLIOGRAFÍA.....	109
6.2. Código MATLAB.....	111

INDICE DE FIGURAS Y TABLAS

LISTA DE FIGURAS

Figura 2.01: Edificio inteligente.....	12
Figura 2.02: Tipos de desvanecimientos	13
Figura 2.03: Relación potencia señal–distancia producida por la atenuación	14
Figura 2.04: Refracción	15
Figura 2.05: Reflexión.....	15
Figura 2.06: Principio de Huygens	16
Figura 2.07: Dispersión	16
Figura 2.08: Respuesta al impulso de un canal multitrayecto	17
Figura 2.09: Respuesta en frecuencia del canal.....	18
Figura 2.10: Canal doblemente selectivo	20
Figura 2.11: Efecto Doppler	21
Figura 2.12: Canal dispersivo en frecuencia	21
Figura 2.13: Tecnologías de localización	22
Figura 2.14: Satélites GPS.....	23
Figura 2.15: OFDM	27
Figura 2.16: Canales 802.11	27
Figura 2.17: Sistema Aeroscout.....	28
Figura 2.18: Mapa de potencia obtenido por Ekahau	28
Figura 2.19: Mota Cricket - 433 MHz	29
Figura 2.20: Posicionamiento de robots por infrarrojos	31
Figura 2.21: Sistema Zonith	32
Figura 2.22: Arquitectura del sistema de Ubisense	33
Figura 2.23: Sistema domótico con una red de sensores Zigbee	33
Figura 2.24: Comparativa de tecnologías	34
Figura 2.25: Clasificación de técnicas de localización.....	36
Figura 2.26: Cálculo de la distancia entre transmisor y receptor en TDOA	37
Figura 2.27: TDOA por comparación por pares de estaciones base	37
Figura 2.28: Sistema de localización por ángulo de llegada	38
Figura 3.01: Supertrama Zigbee	41
Figura 3.02: Diagrama de tiempos TDOA de un sistema unilateral	42
Figura 3.03: Diagrama de tiempos TDOA de un sistema multilateral	43
Figura 3.04: Localización TDOA en 2 dimensiones	43
Figura 3.05: Indicadores CEP y DRMS gráficamente	46
Figura 3.06: Escenario posicionamiento del método de Fang.....	50
Figura 3.07: Gráfico de notación del método de interpolación esférica.....	54
Figura 3.08: Localización en un escenario 2D	57
Figura 3.09: Notación del método de Friedlander	61
Figura 3.10: Método de Steepest Descent	66

Figura 3.11: Variación del valor de theta en el método de Steepest Descent.....	68
Figura 3.12: Ejemplo de mapa de potencias en una vivienda	70
Figura 3.13: Patrón de radiación de antena omnidireccional	71
Figura 3.14: Ejemplo de red bayesiana del sistema Nibble.....	74
Figura 3.15: Perceptrón multicapa con una capa intermedia para localización 2D	75
Figura 3.16: Métodos de localización implementados	76
Figura 3.17: Error medio de los diferentes algoritmos	77
Figura 3.18: Rangos de error en el escenario grande	78
Figura 3.19: Error medio vs SNR	79
Figura 3.20: Dependencia de las estimaciones con el punto inicial de iteración	79
Figura 3.21: Escenario con varios objetos.....	80
Figura 3.22: Mapa de potencia utilizando dos antenas WiFi	82
Figura 3.23: Escenario Carlos III	82
Figura 3.24: Escenario Jerónimo de Ayanz I	83
Figura 3.25: Escenario Jerónimo de Ayanz II	83
Figura 3.26: Escenario SB	84
Figura 3.27: Escenario HVC	84
Figura 3.28: Escenario LTE.....	85
Figura 3.29: Error medio según el escenario utilizando 4 antenas.....	90
Figura 3.30: Error medio según el escenario utilizando 5 antenas	90
Figura 3.31: Errores medios por algoritmo	91
Figura 3.32: Recorrido sobre el plano del escenario Carlos III.....	92
Figura 3.33: Recorrido obtenido con el algoritmo de Chan	93
Figura 3.34: Recorrido obtenido con el algoritmo de Levenberg.....	93
Figura 3.35: Recorrido obtenido con el algoritmo de Steepest Descent	94
Figura 3.36: Comparación de resultados de TDOA y RSS	95

LISTA DE TABLAS

Tabla 2.1: Efecto del multitrayecto en tiempo y frecuencia.....	19
Tabla 2.2: Efecto de la variabilidad en el tiempo de la respuesta del canal	22
Tabla 2.3: Estado actual de los sistemas RFID	30
Tabla 2.4: Resumen de tecnologías	35
Tabla 2.5: Comparación de técnicas de localización.....	39
Tabla 3.01: Resumen de algoritmos de estimación	69
Tabla 3.02: Ejemplo de radio mapa patrón.....	72
Tabla 3.03: Parámetros del algoritmo del trazado de rayos	81
Tabla 3.04: Carlos III 4 antenas.....	86
Tabla 3.05: Carlos III 5 antenas.....	86
Tabla 3.06: Jerónimo de Ayanz I 4 antenas.....	86
Tabla 3.07: Jerónimo de Ayanz I 5 antenas.....	86
Tabla 3.08: Jerónimo de Ayanz II 4 antenas	87

Tabla 3.09: Jerónimo de Ayanz II 5 antenas	87
Tabla 3.10: HVC 4 antenas.....	87
Tabla 3.11: HVC 5 antenas.....	87
Tabla 3.12: LTE 4 antenas.....	88
Tabla 3.13: LTE 5 antenas.....	88
Tabla 3.14: SB 4 antenas	88
Tabla 3.15: SB 5 antenas	88
Tabla 3.16: Dimensiones de los escenarios	89
Tabla 3.17: Porcentajes de convergencia medios	92
Tabla 3.18: Resultados obtenidos para el seguimiento de un individuo	93
Tabla 3.19: Parámetros de simulación utilizados para KNN.....	94
Tabla 3.20: Resultados KNN.....	95

Memoria

Introducción

Bloque 1

1. Introducción y objetivos

Los sistemas de localización en interiores han cobrado un interés creciente, por su utilidad para llevar a cabo seguimientos de bienes o personas que permita conocer su posición en tiempo real. Estos sistemas en auge tienen aplicaciones tan diversas como permitir una mejora en la eficiencia en el almacenamiento de productos en una nave industrial, o analizar los patrones de movimiento de personas dentro de un edificio por motivos de seguridad.

Estos sistemas no han alcanzado todavía el éxito de los sistemas de localización en exteriores como el GPS por dos razones principalmente; complejidad y coste. La localización en interiores plantea retos tecnológicos mucho más complejos que los problemas a los que se enfrentan los sistemas en entornos abiertos. Para enfrentarse a estas dificultades, es necesaria una gran infraestructura con un número inmenso de sensores o antenas incrementando el coste de estas instalaciones.

Una forma de paliar la cuestión económica se consigue aprovechando redes existentes como las redes 802.11, ampliamente extendidas, aunque de esta forma no se alcanzan altas precisiones. Existe un compromiso de precisión frente a coste que debe ser estudiado detenidamente en cada caso para ver cuál es la solución más adecuada.

Existen multitud de sistemas comerciales y en investigación basados en localización en interiores que abarcan el abanico de tecnologías inalámbricas existentes en el mercado. Estos sistemas están orientados al propósito general y por tanto son poco adaptables a necesidades concretas. La elección prioritaria del usuario consiste en decidir la tecnología o tecnologías, en caso de elegir un sistema híbrido, que va a utilizar porque dicha decisión afectará directamente al coste y a la precisión máxima a alcanzar.

Tal como se ve, existe una gran diversidad en el campo de la localización en interiores y esto ha provocado ser el tema de interés de este proyecto. Con la realización de este proyecto se pretende mostrar una visión más clara de las técnicas matemáticas empleadas en este campo para comprender los problemas a los que se enfrentan los sistemas de localización indoor.

1.1. OBJETIVOS

Los objetivos del proyecto, tal como se muestra en el título del mismo, consisten en el estudio de los sistemas de localización indoor y la implementación de los sistemas más versátiles.

En una primera parte se realizará un estudio de las diferentes técnicas existentes para la localización. Una vez conocido el estado actual se implementarán los algoritmos de localización más utilizados.

El objetivo principal es la realización de una comparación principalmente en cuanto a precisión de los algoritmos implementados.

Como ayuda para conseguir dicho objetivo se utilizará un algoritmo de trazado de rayos que proporcionará medidas simuladas de escenarios reales. En este caso, las medidas de interés serán los valores de potencias y retardos en los diferentes puntos de los escenarios simulados.

1.2. ESTRUCTURA DEL PROYECTO

El proyecto comienza con una breve introducción y posteriormente está dividido en 5 bloques descritos a continuación:

- Dentro del bloque 2: Estado del arte, se describen las tecnologías y los sistemas actuales existentes en cuanto a localización ya sea en interior o en exterior.
- A continuación, en el bloque 3: técnicas y sistemas de localización, se describen matemáticamente los algoritmos de localización indoor en los que se ha fundamentado el proyecto. Posteriormente se muestran los resultados obtenidos tras la implementación y prueba de los algoritmos.
- En el bloque 4 se describe un proyecto real fundamentado en los sistemas de localización mostrando el estado actual del mismo y los resultados a los que se han llegado.
- En el bloque 5 aparecen las conclusiones obtenidas tras la realización del proyecto y se muestran las líneas de estudio que pueden ser llevadas a cabo en el futuro.
- Finalmente en el bloque 6 se muestran los anexos del proyecto en los que se describe la bibliografía utilizada para el desarrollo del proyecto y los códigos de Matlab más importantes desarrollados para este proyecto.

Memoria

Estado del arte

Bloque 2

2. Estado del arte

En este capítulo se introduce el problema de la localización y se realiza una clasificación general de los sistemas que permiten estimar la posición de un objeto móvil en interiores.

El rápido avance de la tecnología, así como de los sistemas inteligentes ha hecho posible el desarrollo de sistemas con funciones de localización, es decir, con la capacidad de determinar la ubicación del dispositivo en un área determinada, para diversas funcionalidades. Tenemos dos tipos de sistemas, sistemas de localización en los que el usuario o terminal es el encargado de obtener su propia posición (como por ejemplo el sistema navegador de los vehículos o la orientación de usuarios móviles), y sistemas de gestión centralizados (seguimiento de usuarios en las redes, móviles o WLAN).

La investigación destinada a la localización en interiores se origina tras el fracaso de los sistemas GPS de localizar dentro de un edificio de manera adecuada. La localización en interiores tiene un futuro prometedor en el ámbito de la Inteligencia Ambiental (IA), que es una disciplina en la que se pretende que una persona use en su vida cotidiana múltiples dispositivos de computación que le faciliten la misma. En la IA se pretende conseguir una integración de tecnologías enfocadas hacia vehículos inteligentes, edificios inteligentes, etc.



Figura 2.1: Edificio inteligente

Un ejemplo de edificio inteligente en el cual es útil un sistema de localización es un museo. La idea sería asignarle a cada visitante un objeto que le permitirá recibir detalles que no están escritos en los paneles informativos de las obras de arte. Este sistema de localización necesitaría conocer la localización del usuario así como su

orientación, y quizá otros detalles como su idioma materno, y al presionar un botón reproduciría la información de la obra de arte que está mirando el sujeto.

Vamos a comenzar explicando los efectos que se producen sobre las señales afectando a los sistemas de localización. Una vez que se conozcan estos efectos, se explicaran las diferentes tecnologías y técnicas que existen para el posicionamiento en interiores.

2.1. CANAL RADIO

Las ondas de radio son ondas que poseen una componente eléctrica y una componente magnética y como tales, están expuestas a ciertos fenómenos capaces de modificar el patrón de propagación de las mismas.

En la literatura, a menudo se clasifican los efectos del canal inalámbrico sobre la señal en dos grandes grupos:

- Desvanecimientos a gran escala
- Desvanecimientos a pequeña escala

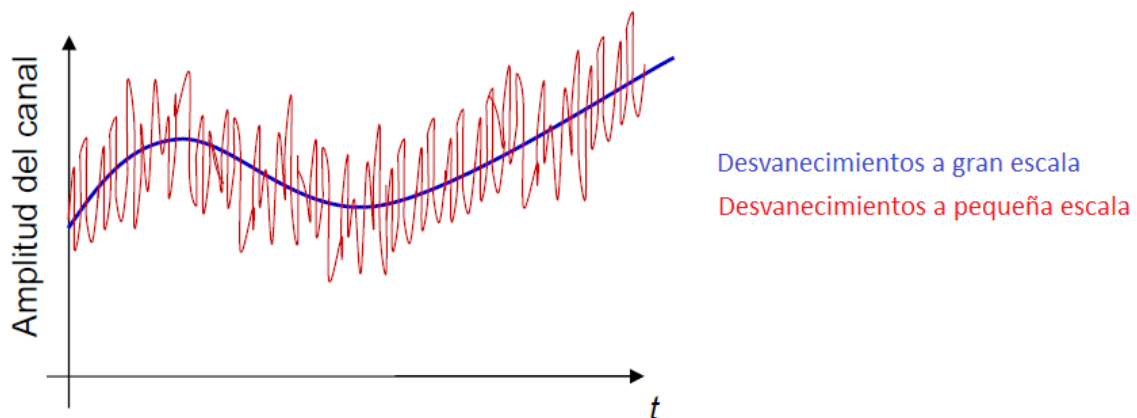


Figura 2.2: Tipos de desvanecimientos

2.1.1. Desvanecimientos a gran escala

Los efectos de gran escala se deben a la obstaculización de la señal por grandes objetos del entorno de propagación, tales como edificios, montañas,..., así como a la atenuación que sufre la señal con la distancia entre transmisor y receptor. En este caso, los cambios temporales que experimenta el canal son debidos a los movimientos del receptor (o transmisor) que involucran distancias muy superiores a la longitud de onda de operación del sistema, por lo que la variación temporal del canal se produce muy lentamente en comparación con el periodo de símbolo del sistema.

La pérdida de propagación del trayecto, medida en unidades logarítmicas, se compone de una componente media constante y una componente aleatoria con función densidad de probabilidad log-normal que modela la variación de la pérdida del trayecto debida al movimiento aleatorio del target en las distancias anteriormente comentadas.

2.1.1.1. Atenuación

La atenuación es el efecto por el cual la potencia de la señal transmitida disminuye con la distancia y esta disminución es mayor al aumentar la frecuencia de operación. Existen diferentes modelos de cálculo de atenuación dependiendo de la orografía del escenario ya que por ejemplo no se producen las mismas pérdidas en interiores que en exteriores.

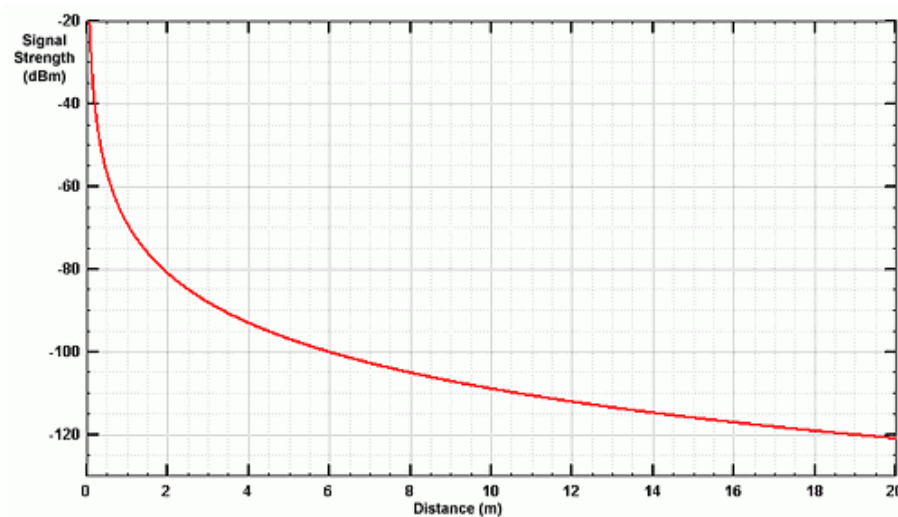


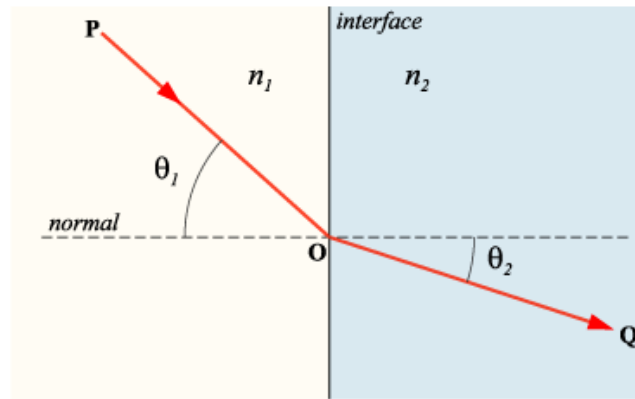
Figura 2.3: Relación potencia señal–distancia producida por la atenuación

2.1.1.2. Refracción

Las ondas de radio están expuestas a sufrir una desviación en su trayectoria cuando atraviesan de un medio a otro con densidad distinta. En comunicaciones, este efecto sucede en casos tan dispares como cuando las ondas electromagnéticas atraviesan las distintas capas de la atmósfera variando su trayectoria en un cierto ángulo o las paredes.

La desviación de la trayectoria es proporcional al índice de refractividad, el cual depende de la velocidad de propagación en el espacio libre V_p y la velocidad de propagación en el medio V_m .

$$\text{Índice Refracción} = \frac{V_p}{V_m}$$



$$n_1 * \text{sen}\theta_1 = n_2 * \text{sen}\theta_2$$

Figura 2.4: Refracción

2.1.1.3. Reflexión

La reflexión se produce cuando la onda electromagnética colisiona sobre un medio de diferentes características. Si el medio es dieléctrico parte de esa onda se refleja y parte de transmite; mientras que en un conductor perfecto, la reflexión es total. Las reflexiones se producen sobre la superficie de la tierra, edificaciones, paredes o incluso con la ionosfera.

El coeficiente de reflexión depende del material, la polarización de la onda, el ángulo de incidencia y la frecuencia.

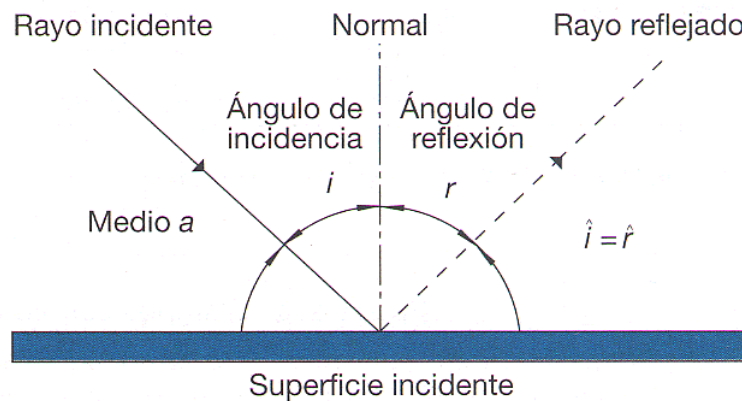


Figura 2.5: Reflexión

2.1.1.4. Difracción

Este fenómeno se produce cuando el camino entre el emisor y el receptor está obstruido por una superficie que tiene irregularidades agudas o puntas. Este fenómeno se puede explicar a partir del principio de Huygens en el que se afirma que todo punto del frente de onda es una fuente aunque un mecanismo para simplificar el estudio es con la geometría de las zonas de Fresnel y el modelo del filo de cuchillo.

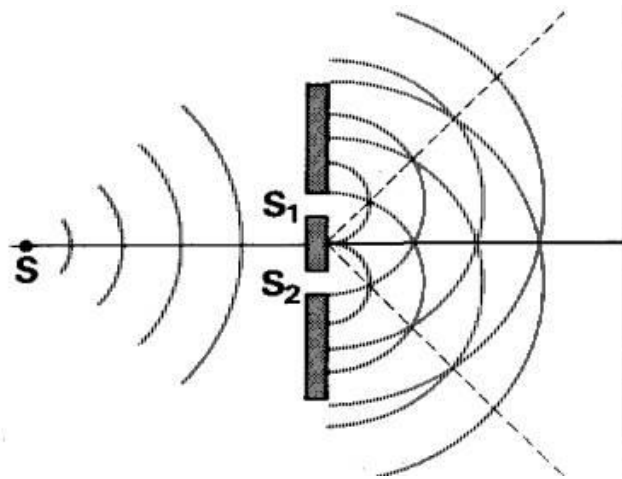


Figura 2.6: Principio de Huygens

La onda secundaria es el resultado de la obstrucción a lo largo de la superficie incluso detrás del obstáculo dando lugar a un doblamiento de las ondas alrededor del obstáculo incluso cuando no existe trayectoria de línea de vista entre el transmisor y el receptor. A altas frecuencias, la difracción, como la reflexión, depende de la geometría del objeto, así como la amplitud, fase y polarización de la onda incidente al punto de difracción.

2.1.1.5. Dispersión

La dispersión ocurre cuando el medio por el cual viaja la onda está formado por objetos con dimensiones pequeñas comparadas a la longitud de onda, y donde hay un gran número de obstáculos por volumen de unidad.

Si hay muchos objetos en la trayectoria de la señal, y los objetos son relativamente pequeños comparados con la longitud de onda de la señal, entonces el frente de onda propagado se dividirá en muchas direcciones. La señal resultante se dispersará en todas las direcciones añadiéndose a las interferencias constructivas y destructivas de la señal.

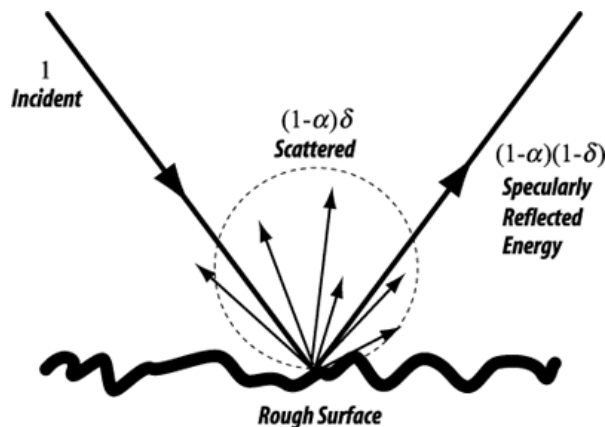


Figura 2.7: Dispersión

2.1.2. Desvanecimientos a pequeña escala

Los efectos de pequeña escala, por su parte, se deben a las interferencias constructiva y destructiva de las señales recibidas por múltiples trayectos en el receptor. La variabilidad temporal del canal se debe a los movimientos del receptor (o transmisor), dependiendo de la arquitectura del escenario de localización, que involucran distancias del orden de la longitud de onda de operación del sistema. Estos efectos se modelan a través de la respuesta impulsiva del canal o, de forma equivalente, de la respuesta en frecuencia del mismo.

2.1.2.1. Propagación multitrayecto

En las transmisiones inalámbricas la señal transmitida normalmente llega hasta el receptor a través de múltiples trayectos, cada uno de los cuales tiene cierto retardo y cierta atenuación, en general, diferentes.

Vamos a considerar que el escenario se mantiene estacionario (todos los elementos del mismo, incluyendo transmisor y receptor, permanecen fijos), por lo que el canal será invariante en el tiempo y se podrá modelar en banda base como un tren de deltas de distinta amplitud y distinto retardo. De esta forma, la señal recibida se podrá expresar como la suma de las réplicas de la señal transmitida, retrasadas y atenuadas, que aparecen a la salida del canal debido a los múltiples trayectos que este presenta, tal y como se ilustra en la figura 2.8. En dicha figura también se refleja el hecho de que un canal multitrayecto es un canal dispersivo en el tiempo.

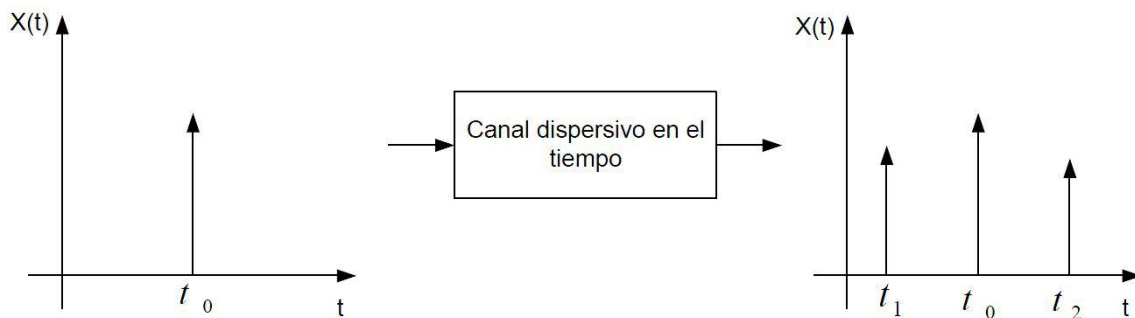


Figura 2.8: Respuesta al impulso de un canal multitrayecto

En cuanto a la distribución de la potencia del canal entre los trayectos resolubles que lo forman, lo habitual será que ciertos trayectos sufran una mayor atenuación que otros, por lo que las réplicas de la señal que se transmiten por cada uno de ellos llegarán al receptor con distinta potencia.

Para tener en cuenta este aspecto, se utiliza el denominado perfil potencia-retardo del canal, el cual determina la forma en la que se distribuye la potencia del mismo entre los distintos trayectos que lo componen. Los dos perfiles más utilizados son el perfil uniforme (todos los trayectos tienen la misma potencia) y el perfil exponencial.

En el perfil exponencial los trayectos de mayor retardo tienen una menor potencia que los de menor retardo, por lo que las componentes de la señal que se transmitan por aquellos sufrirán una mayor atenuación. Esto es lo que suele ocurrir en la práctica.

Llegados a este punto podemos ver que el primer efecto que produce el canal es un ensanchamiento temporal en la señal recibida como consecuencia de los distintos retardos que experimenta la señal transmitida al atravesar cada uno de los trayectos del canal. Este ensanchamiento de los símbolos recibidos provocará que símbolos adyacentes se puedan solapar dando lugar a la aparición de interferencia intersímbolo o ISI. Para medir la importancia de este efecto, se define el ensanchamiento del retardo del canal, T_n , como el máximo retardo del mismo. Mientras menor sea el periodo de símbolo, T_s , frente a este parámetro, mayor será la probabilidad de interferencia entre símbolos cercanos.

En cualquier sistema de comunicaciones, la existencia de ISI supone un aumento en la tasa de error de bit. Para que la tasa de error de bit no aumente en exceso, es importante mantener un nivel bajo de ISI, para lo cual es necesario limitar la tasa del sistema para que el periodo de símbolo sea lo suficientemente grande en comparación con el ensanchamiento del retardo. De hecho, cuando T_n es mucho menor que T_s , los distintos trayectos del canal no serán resolubles, por lo que se podrá considerar que este consta de un único trayecto lo que implicará que el canal no introduce ningún tipo de ISI.

El efecto que el multitrayecto del canal tiene sobre la señal transmitida también se puede apreciar en el dominio de la frecuencia. Para ello se analiza la respuesta en frecuencia del canal y se puede observar que el módulo de la función de transferencia no es constante con la frecuencia.

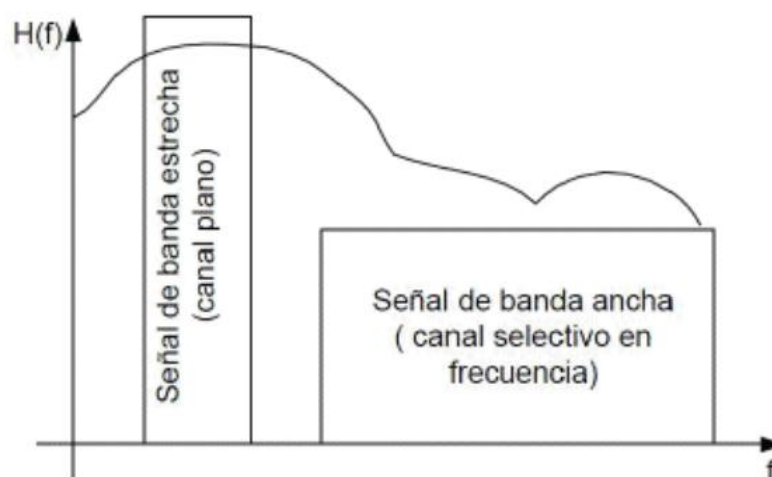


Figura 2.9: Respuesta en frecuencia del canal

Para analizar el efecto que esta dependencia frecuencial tiene en nuestra señal, se define el ancho de banda de coherencia del canal, B_c , como aquel en el cual la respuesta

en frecuencia del canal permanece aproximadamente constante. Si el ancho de banda de la señal, B_s , es menor que este parámetro, entonces el canal podrá considerarse no selectivo en frecuencia (o plano en frecuencia) y por tanto no distorsionará el espectro de nuestra señal. En caso contrario, el canal se comportará como selectivo en frecuencia y distorsionará el espectro de la misma.

Si queremos relacionar ambos modelos, el ancho de banda de coherencia y el ensanchamiento del retardo del canal se pueden relacionar de forma aproximada por la expresión:

$$B_c \approx \frac{1}{T_n}$$

DOMINIO DEL TIEMPO	DOMINIO DE LA FRECUENCIA
T_s grande frente a T_n ↓ Trayectos no resolubles Canal no dispersivo en el tiempo	B_s pequeño frente a B_c ↓ Canal plano o no selectivo en frecuencia
T_s pequeño frente a T_n ↓ Trayectos resolubles ↓ Aumenta la ISI Canal dispersivo en el tiempo	B_s grande frente a B_c ↓ Canal selectivo en frecuencia

Tabla 2.1: Efecto del multitrayecto en tiempo y frecuencia

2.1.2.2. Efecto Doppler

En el escenario de propagación anterior consideramos los elementos en posiciones fijas. Sin embargo, en ciertos escenarios prácticos debemos considerar también el posible movimiento de estos elementos (sobre todo del target). Esto va a provocar que el canal tenga una cierta variabilidad temporal ya que la posición de los elementos cambiará con el tiempo y por tanto así lo harán las fuerzas del canal. A los canales que son selectivos tanto en tiempo como en frecuencia se les suele denominar canales doblemente selectivos.

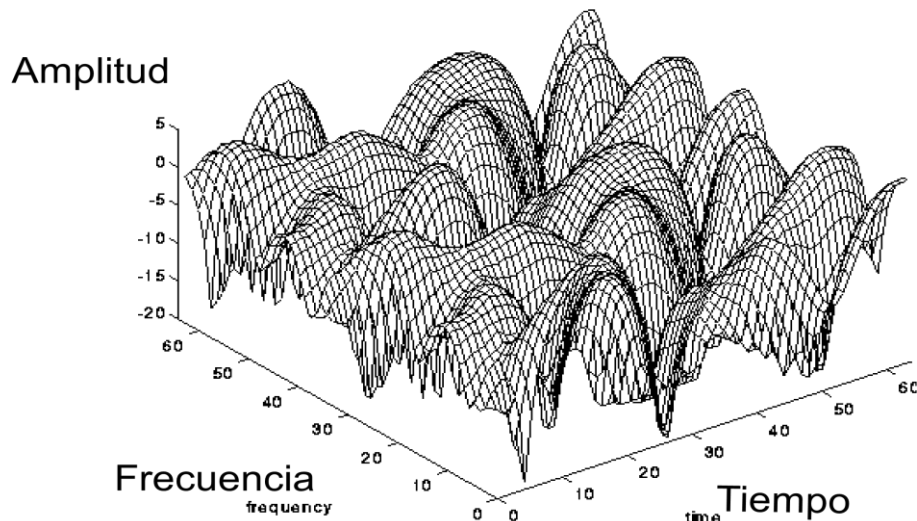


Figura 2.10: Canal doblemente selectivo

Para medir los efectos que la variabilidad temporal produce en nuestra señal, vamos a definir dos parámetros. En primer lugar, definimos el tiempo de coherencia del canal, T_c como la longitud del intervalo de tiempo en el cual podemos considerar que la respuesta del canal no cambia. De esta forma, si $T_s \ll T_c$, el canal no cambiará significativamente durante la transmisión de un símbolo y hablaremos de desvanecimiento lento (o de no selectividad en el tiempo). En caso contrario el canal sí cambiará durante la transmisión de un símbolo y hablaremos de desvanecimiento rápido (o de selectividad en el tiempo).

La variabilidad temporal del canal está íntimamente relacionada con el conocido efecto Doppler, el cual se debe al hecho de que haya elementos en movimiento en el escenario de propagación. Este efecto se va a traducir en un ensanchamiento del espectro de la señal recibida y en la aparición de un offset de frecuencia que desplazará el espectro de la señal recibida respecto al de la transmitida.

El desplazamiento Doppler es de sobra conocido. Consiste en un cambio en las frecuencias recibidas respecto a las transmitidas cuando existe un movimiento relativo entre transmisor y receptor. Esto se traduce en un desplazamiento de la frecuencia de portadora percibida por el receptor. El ensanchamiento Doppler se produce debido a que el movimiento relativo entre transmisor y receptor, para cada uno de los trayectos, puede ser distinto por lo que las componentes que hayan viajado por cada uno de los trayectos habrán sufrido un desplazamiento Doppler diferente.

En la figura 2.11 se representa gráficamente este concepto apareciendo con línea discontinua la señal transmitida y con línea continua la señal a la salida del canal. Aparecen varias variables como f_d que representa al desplazamiento Doppler y B_d al ensanchamiento Doppler. El ensanchamiento Doppler, B_d , está relacionado con el tiempo de coherencia del canal, T_c , pudiéndose expresar aproximadamente como:

$$B_d \approx \frac{1}{T_c}$$

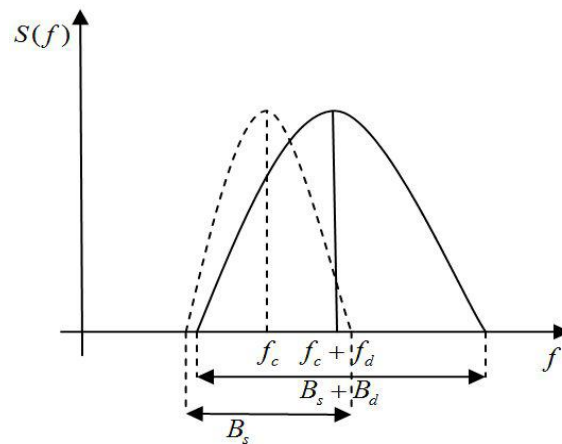


Figura 2.11: Efecto Doppler

Si transmitiéramos a través del canal una señal monocromática, debido al ensanchamiento Doppler, a la salida del mismo aparecerán nuevas componentes espectrales que no estaban presentes a la entrada, como se puede apreciar en la figura 2.12. Es por esto por lo que se dice que estos canales son dispersivos en frecuencia.

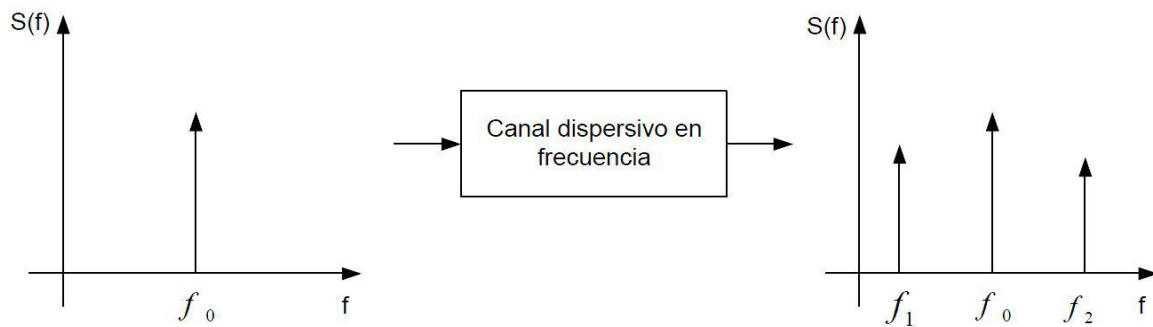


Figura 2.12: Canal dispersivo en frecuencia

Como veremos, el ensanchamiento y desplazamiento Doppler son dos de los principales causantes de la aparición de interferencia entre canal o ICI en los sistemas multiportadora, por lo que hay que tenerlo muy en cuenta a la hora de diseñar el sistema.

En la tabla 2.2 se recoge un resumen del efecto que la variabilidad en el tiempo de la respuesta del canal (o, equivalentemente, el efecto Doppler) tiene sobre la señal que se transmite por él.

DOMINIO DEL TIEMPO	DOMINIO DE LA FRECUENCIA
T_s grande frente a T_c ↓ Desvanecimientos rápidos	B_s pequeño frente a B_d ↓ Efecto Doppler no despreciable (Canal dispersivo en frecuencia)
T_s pequeño frente a T_c ↓ Desvanecimiento plano	B_s grande frente a B_c ↓ Efecto Doppler despreciable (Canal no dispersivo en frecuencia)

Tabla 2.2: Efecto de la variabilidad en el tiempo de la respuesta del canal

2.2. TECNOLOGÍAS PARA LA LOCALIZACIÓN

En la siguiente tabla se muestran las principales tecnologías de localización que serán explicadas a continuación. Se han incluido las técnicas *outdoor* para mostrar la diversidad completa de soluciones existentes.

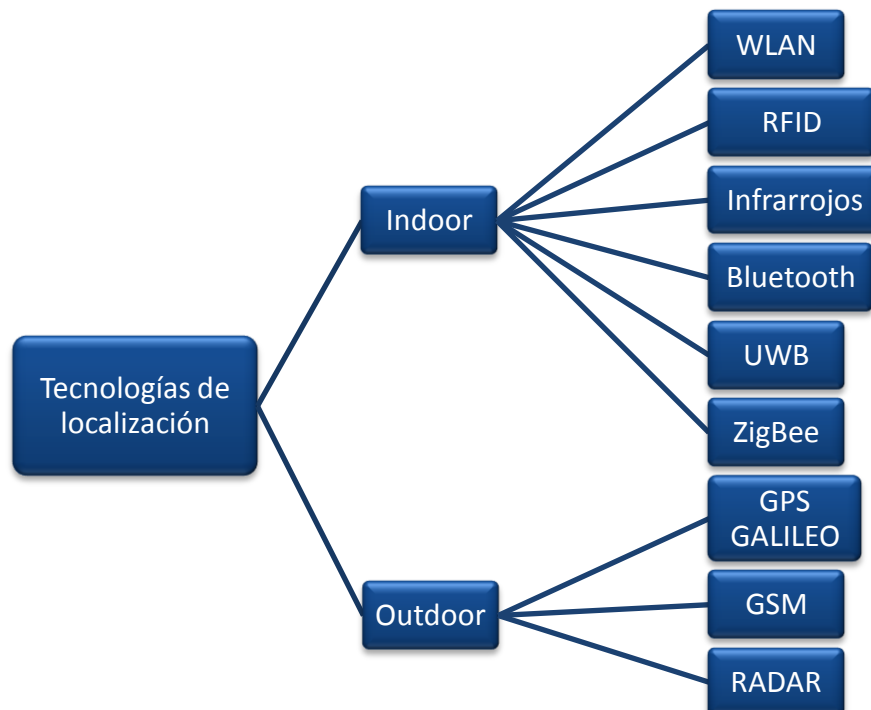


Figura 2.13: Tecnologías de localización

2.2.1. GPS

El sistema de posicionamiento pionero es el conocido como Sistema de Posicionamiento Global (GPS – Global Positioning System). Fue desarrollado en el ámbito militar (por la empresa NAVSTAR), como herramienta para la estimación de Posición, Velocidad y Tiempo de vehículos. Entró en servicio en el año 1978. Poco tiempo después surgió el sistema ruso GLONASS, de similares características, también para uso militar. A pesar de su origen militar, paulatinamente han ido apareciendo aplicaciones civiles, que han tenido un crecimiento espectacular en los últimos años como la navegación de vehículos por carreteras.

Los localizadores por GPS reciben el soporte de una constelación de hasta 24 satélites, que orbitan por todo el globo terrestre enviando sus señales a todo aquel que quiera oírlas. Un receptor de GPS que quiere localizarse dentro del globo terrestre localiza al menos a cuatro satélites – cuanto mayor sea el número de satélites encontrado mejor será su estimación de la posición – y de cada uno de ellos obtiene la posición del satélite emisor y el tiempo de envío de cada muestra recibida. Con estos datos, el receptor GPS calcula por triangulación su posición absoluta dentro de la tierra (latitud, longitud, altitud) gracias a que los satélites emiten en el mismo preciso momento su señal, pero ésta le llega retardada al receptor por razones obvias de distancia.

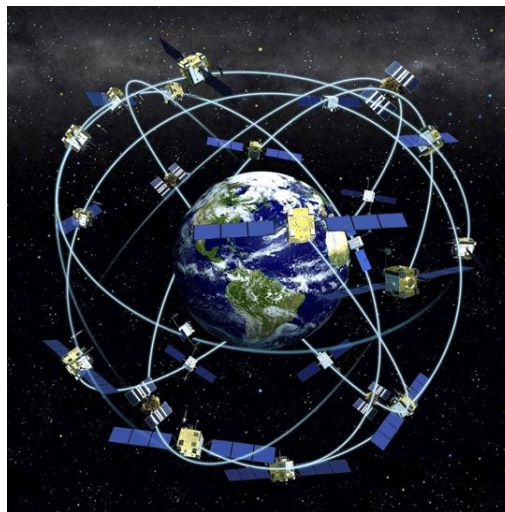


Figura 2.14: Satélites GPS

La arquitectura GPS mostrada en la figura 2.14 tiene una precisión que puede llegar a ser mejor de 10 metros si se toman en consideración más de cuatro satélites, lo cual es bastante interesante para localizar en el mundo a nivel global, pero no tiene mucho sentido usarlo para localización dentro de un área pequeña, como puede ser un edificio.

Otro inconveniente adicional que nos encontramos con el GPS es que, al necesitar línea de visión directa (LoS), hay veces que no se encuentran suficientes satélites al alcance para permitir una localización correcta, esto puede ocurrir en ciudades con rascacielos o calles demasiado estrechas y en túneles. Además, normalmente se requerirá que el dispositivo tenga grabado un mapa porque, sin él, sólo se podrá informar al usuario de su longitud, latitud y altitud, y estos datos no suelen ser suficientes para los usuarios.

Además, las señales del GPS viajan muchos kilómetros y son bastante tenues, por lo que un receptor GPS en el interior de un edificio lo tiene muy complicado para encontrar señales procedentes de los satélites y más aún para conseguir que estas señales le sirvan para localizarse. No obstante, el GPS, como su nombre indica, es un sistema de posicionamiento global, y no está por tanto indicado para permitir la localización a nivel local.

Otra última diferencia básica entre lo que ofrece el GPS y lo que debería ser un sistema de localización en interiores, es que, en un edificio, la referencia, más que una referencia absoluta como la que da el GPS, debería ser una habitación, una zona de una habitación, etc. Por tanto, es preferible que se trate de una referencia relativa, principalmente debido a que dentro de un edificio conocido, a cualquier persona le resultará más informativa una frase del tipo “estás en el despacho de Juan” que otra como “tu situación en el espacio es (X, Y, Z)”.

Debido a la inaccesibilidad en los entornos urbanos e interiores, se ha desarrollado una serie de extensiones de sistema GPS, para obtener un mejor rendimiento. Estas extensiones se apoyan en otras infraestructuras de red que complementan a los satélites:

- Utilización de las redes móviles para enviar información adicional sobre las órbitas de los satélites y correcciones de las señales a través de un receptor de datos. Esta técnica se denomina GPS Asistido (A-GPS, Assisted Global Positioning System).
- El D-GPS (Differential GPS), o GPS diferencial, es un sistema que proporciona a los receptores de GPS correcciones de los datos recibidos de los satélites GPS, con el fin de proporcionar una mayor precisión en la posición calculada.
- Otras técnicas mejoradas para aumentar la precisión a pocos metros, y algunos incluso a unos pocos milímetros.

2.2.2. Galileo

Galileo es el equivalente europeo al GPS, que se está desarrollando en estos momentos y del cual hay dos satélites en prueba orbitando; Giove-A desde 2005 y Giove-B desde 2008. Se espera su puesta en marcha para el año 2014 y estará destinado al uso civil. Es un Sistema Global de Navegación por Satélite (GNSS) que evitará la

dependencia de los sistemas GPS y GLONASS, aunque será un sistema interoperable con estos.

Proporcionará servicios de radionavegación así como de posicionamiento en el espacio, dividido en 5 servicios:

- Servicio abierto (Open Service – OS), servicio público gratis.
- Servicio para aplicaciones críticas (Safety of Life – SoL)
El SoL mediante pago está pensado para sistemas de aviación, proporcionando más integridad (es un sistema que avisa de la fiabilidad de las medidas).
- Servicio público regulado (Public Regulated Service – PRS) para policía (con mayor integridad y precisión).
- Servicio de búsqueda y salvamento (Search and Rescue Service – SAR).
- Servicio Comercial (Commercial Service – CS)

El receptor usará satélites de distintas constelaciones para el cálculo de su ubicación en el espacio. Proporcionará ubicación en tiempo real con una precisión de metros gracias a las dos frecuencias en las que trabaja en su versión estándar.

El sistema Galileo estará formado por una constelación mundial de 30 satélites en órbita terrestre media distribuidos en tres planos inclinados con un ángulo de 56° hacia el ecuador, a 23.616 Km. de altitud. Estos satélites se encuentran ligeramente más inclinados hacia los polos que los de la malla GPS para proporcionar mejor cobertura en áreas cercanas a los polos pues allí es donde menos cobertura proporciona GPS.

2.2.3. 802.11

La comunicación típica en el protocolo 802.11 sigue un modelo centralizado. Por tanto, una red consta de uno o varios puntos de acceso (APs) y multitud de clientes conectados a cada uno de los puntos de acceso. Cada AP emite periódicamente una baliza o beacon para hacer notar su presencia a los usuarios (representados por tarjetas de red inalámbricas), los cuales de este modo pueden saber en todo momento qué redes inalámbricas hay disponibles en su entorno.

Las características principales de los protocolos 802.11 son las siguientes:

- Siguen en líneas generales el estándar IEEE.802.3, estándar de redes de área local.
- Usan CSMA como mecanismo de acceso al medio.
- Como banda de frecuencia usan la regulada en torno a los 2.4GHz, en Europa, y 5 GHz en EEUU, para aplicaciones industriales, científicas y médicas.
- Interfaz de tecnología radio:

Utiliza técnicas de modulación de espectro ensanchado (Spread Spectrum). Dichas técnicas emplean una baja densidad de potencia, por lo que la señal no interfiere con otros receptores y a su vez incorporan redundancia, por lo que proporciona a estos canales resistencia a interferencias y al ruido. Existen fundamentalmente varias técnicas:

- FHSS (Frequency Hopping Spread Spectrum):

El código ensanchado de los sistemas FHSS, es una secuencia de frecuencias específicas que se utiliza para modificar la frecuencia portadora de forma periódica cubriendo todo el canal asignado. La duración de cada salto está en el rango de los 100 ms. El mensaje se modula sobre la portadora utilizando FSK, lo cual genera una señal de banda estrecha para la duración de cada salto, pero de banda ancha si se observa el canal en intervalos de tiempo de varios segundos.

- DSSS (Direct Sequence Spread Spectrum):

La señal utiliza todo el ancho de banda del canal asignado, durante la duración de cada bit, la portadora se modula mediante PSK, siguiendo una secuencia de bits específica (chip). La portadora se modulará con un determinado chip en función de si el bit de mensaje aparece a cero o a uno, usando la misma secuencia pero invertida.

Si cada usuario utiliza un código chip diferente, el canal podrá ser utilizado por varios usuarios de forma concurrente. Esta técnica se conoce con el nombre de CDMA, (Code Division Multiple Access).

En el 802.11, se utilizan códigos ortogonales de longitud 11, por lo que no es posible utilizar CDMA.

También se emplea otra técnica que no es de espectro ensanchado pero que proporciona un rendimiento bastante superior a la misma velocidad que las de espectro ensanchado. Esta técnica se conoce como OFDM (Orthogonal Frequency Division Multiplexing).

En OFDM, el ancho de banda asignado se divide en canales y la señal a enviar en tramas, de tal forma que cada una de ellas es modulada por la frecuencia de subportadora de cada canal. De esta forma se busca optimizar el canal de comunicaciones, reforzando los canales más débiles, aquellos más ruidosos o atenuados.

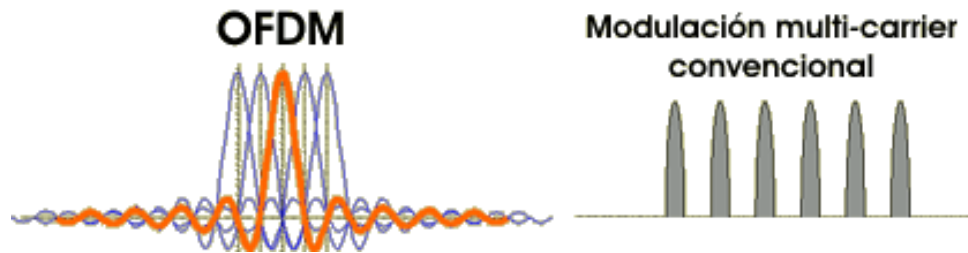


Figura 2.15: OFDM

Las redes basadas en los protocolos 802.11b y 802.11g emiten en uno de los 14 canales solapados entre sí establecidos internacionalmente, aunque no todos ellos son válidos en todos los países (en España en principio sólo se permitía el uso de 2 canales, aunque actualmente se permiten los canales del 1 al 13). Cada uno de los canales tiene un espectro de frecuencias como el mostrado en la figura 2.16. El canal de emisión de la red es normalmente configurable por parte del usuario pero no se puede establecer de forma automática dependiendo de la distinta carga de los canales, por lo que las redes emisoras podrían interferir entre sí; es por tanto responsabilidad del técnico instalador de la red asegurarse de no colocar su red en un canal fuertemente utilizado para permitir la mayor fluidez en las comunicaciones, y las recomendaciones típicas son usar los canales que no se solapan entre sí (1, 6 y 11).

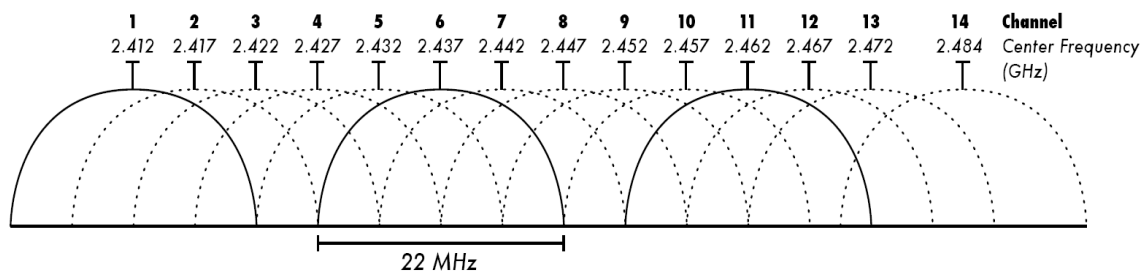


Figura 2.16: Canales 802.11

Existen dos empresas norteamericanas que poseen sistemas comerciales que utilizan esta tecnología como medio de localización; Ekahau y AeroScout.

El sistema de Aeroscout también hace uso de RFID por lo que se puede denominar un sistema híbrido. En este sistema, los terminales que van a ser localizados o targets son los emisores del sistema. Los puntos de acceso reciben las señales y envían los tiempos o las potencias de llegada al motor de procesamiento del sistema. Tras calcular la posición, esta se indica de manera gráfica para el usuario.

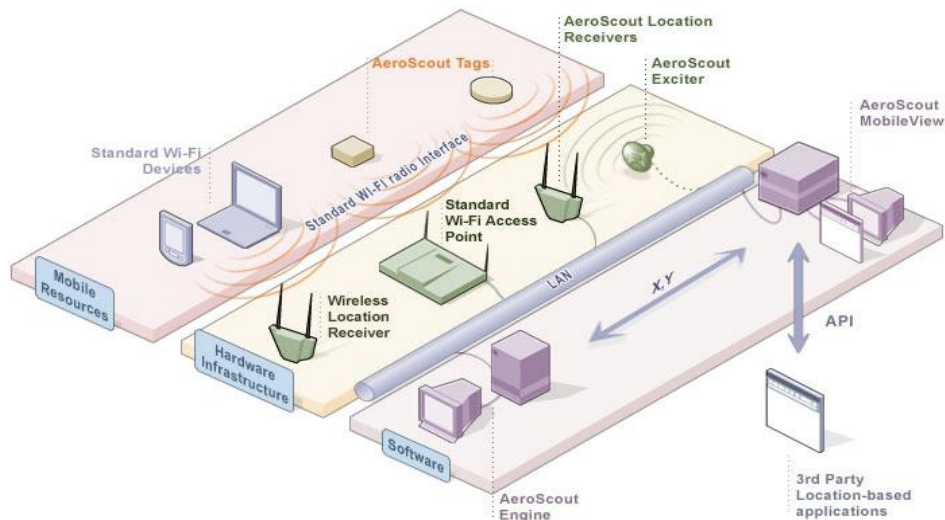


Figura 2.17: Sistema AeroScout

Por otro lado, el sistema Ekahau es bastante más básico. Funciona tanto en interiores como en exteriores, únicamente se necesita cobertura en una red local inalámbrica (WLAN). Es un sistema plenamente software que se basa solamente en potencias de señal recibidas; sin embargo, se requiere un entrenamiento previo del escenario para su funcionamiento.

Sus principales características son:

- Es compatible tanto con dispositivos de Ekahau pasivos, etiquetas o Tags T201 como con dispositivos activos, ordenadores, PDAs...
- Localiza varios dispositivos simultáneamente sobre un mismo mapa de situación y su precisión es de hasta 1 metro.
- La información que proporciona de cada dispositivo son, coordenadas (x,y), edificio, piso, habitación y zona.
- Funciona sobre cualquier 802.11.
- Cada usuario debe instalarse el software cliente y calibrar los mapas de área para poder comenzar a usar el sistema.



Figura 2.18: Mapa de potencia obtenido por Ekahau

2.2.4. RFID

Otra alternativa en la localización en interiores son los sistemas RFID (Radio Frequency Identification). Este sistema se basa en etiquetas de radiofrecuencia que contienen una antena emisora/receptora que al ser excitada por un transmisor emite una señal. Así, un usuario que se quiera localizar en un edificio necesita tener cerca de él un cierto número mínimo de etiquetas de radiofrecuencia. El propio usuario tendría un lector de etiquetas RFID y leyendo las etiquetas cercanas puede llegar a localizarse utilizando la potencia de señal recibida por las mismas; es decir utilizando la RSS (Received Signal Strength).

Las etiquetas RFID pueden ser activas o pasivas en función de la necesidad de poseer alimentación interna o no. Las etiquetas pasivas son más baratas porque no requieren alimentación pero como consecuencia su rango de alcance es menor. En el caso de etiquetas activas se consiguen potencias de transmisión mayores, menores errores y un alcance aproximado de 100 metros, medida muy superior a las etiquetas pasivas.

Un ejemplo de un sistema de localización que usa RFID es Cricket, un sistema ideado por ingenieros del MIT (Massachusetts Institute of Technology), cuya precisión está entre 1 y 3 cm y ha sido empleado en proyectos como seguimiento de objetos, control de robots o en aplicaciones context-aware (en las cuales la localización del usuario juega un papel muy importante).

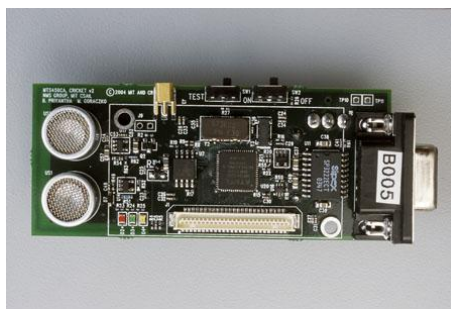


Figura 2.19: Mota Cricket - 433 MHz

En realidad Cricket es un sistema híbrido ya que también hace uso de los ultrasonidos. La comunicación inicial se realiza mediante RFID asegurando fiabilidad en la transmisión de datos pero la respuesta hace uso de los ultrasonidos. Este método funciona correctamente y puede localizar de manera satisfactoria, pero sin embargo es aún un método caro y por esta razón se han seguido buscando otros métodos de localización más baratos.

Como resumen, la localización por RFID es una alternativa válida, sin embargo es muy cara por la gran cantidad de lectores RFID que se necesitan para la localización correcta, ya que el alcance de estas señales es muy reducido y por lo tanto en un

escenario normal se necesitarían cientos de lectores para una localización lo más libre de errores como fuera posible.

Tag	Frecuencia	Rango de alcance	Precisión	Directividad	Memoria de usuario
Pasivo	LF (134.2kHz)	< 50 cm	Muy baja	Muy baja	256 byte
Activo	LF (134.2kHz)	< 150 cm	Muy baja	Muy baja	256 byte
Pasivo	HF (13.56 MHz)	< 1 m	Baja	Baja	2 Kbyte
Activo	HF (13.56 MHz)	< 10 m	Baja	Baja	32 Kbyte
Pasivo	UHF (433, 868-915 MHz)	< 3 m	Media	Media	2 Kbyte
Activo	UHF (433, 868-915 MHz)	> 10 m	Media	Media	128 Kbyte
Pasivo	Microondas (2.45-5.8 GHz)	< 3 m	Alta	Alta	2 Kbyte
Activo	Microondas (2.45-5.8 GHz)	> 10 m	Alta	Alta	128 Kbyte

Tabla 2.3: Estado actual de los sistemas RFID

2.2.5. Localización utilizando la telefonía móvil

Otra alternativa posible, que además no necesitaría ningún hardware adicional para el usuario, pasaría por el uso de un teléfono móvil, y de hecho ya hay operadoras de telefonía móvil que ofrecen la opción de localización vía móvil a sus abonados. Sin embargo, la falta de precisión alcanzada hasta ahora, sitúa a esta tecnología en clara desventaja respecto de otras, ya que los sistemas de localización de este tipo no pueden dar precisiones mayores de 50 metros, por lo que no sirven para nada en interiores.

Esto es debido a que la localización con el uso del teléfono móvil (localización por GSM) se basa en la detección de la célula a la que está conectada el móvil, y en zonas urbanas la precisión es de decenas de metros, pero, sin embargo, en las zonas rurales, donde se necesitan menos células para dar servicio a menor población, esta precisión es mucho menor llegando a kilómetros.

Dentro del bloque 4, se explica un proyecto real de alta envergadura ejecutado por varias empresas en España, donde se hace uso de localización de telefonía móvil para exteriores, alcanzando resultados más que aceptables.

2.2.6. Localización por Infrarrojos

La localización por infrarrojos se puede desechar en un primer momento para la localización en interiores por ser de corto alcance (unos dos metros) y porque, además,

se requieren enlaces LoS. Por su corto alcance habría que incluir una cantidad enorme de emisores de infrarrojos, y aún así serían imposibles de detectar ciertas localizaciones por el problema de LoS. Al igual que en el caso de RFID, este sistema hace uso del indicador RSS para estimar la posición del target.

Existe un proyecto llamado WIPS (Wireless Indoor Positioning System) desarrollado en Suecia, que se basa en la existencia de beacons emitidos vía infrarrojos y unos dispositivos denominados SmartBadges que los llevan los usuarios del sistema de posicionamiento para localizarse tanto de forma pública como de forma anónima.

En definitiva, la localización por infrarrojos es bastante inviable por las limitaciones en cuanto a alcance y a LoS por lo que este sistema se utiliza más como detección que como localización. En temas de robótica se utiliza como detector de obstáculos.



Figura 2.20: Posicionamiento de robots por infrarrojos

2.2.7. Bluetooth

El sistema basado en Bluetooth es similar a los anteriores. Consiste en buscar dispositivos cercanos, obtener el indicador de la potencia recibida (RSS) y utilizando técnicas de triangulación obtener la posición. Bluetooth es una tecnología barata pero al igual que con RFID e infrarrojos, son necesarios muchos dispositivos para cubrir una planta por lo que no está casi implantada de forma comercial.

A pesar de sus limitaciones, las compañías danesas BlueTags y Zonith, poseen un sistema de localización y control basado en Bluetooth. Estos sistemas permiten controlar a aquellas personas u objetos, equipados con el Tag en el caso de BlueTags o cualquier dispositivo con Bluetooth activo en el caso de Zonith, situados dentro de un área de cobertura. De esta forma se puede monitorizar y controlar la situación de aquellas personas u objetos dentro de un área determinada.

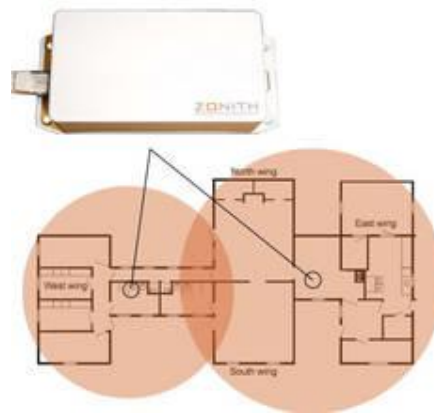


Figura 2.21: Sistema Zonith

2.2.8. UWB (Ultra Wideband)

La tecnología UWB parece una buena candidata para la localización en interiores. Permite altísimas velocidades de transmisión y rangos elevados, y parece un candidato perfecto para sistemas de localización en interiores.

La precisión alcanzada con este sistema es de un metro con lo que quedan patentes sus posibilidades en nuestro ámbito, pero el mayor inconveniente es que no está regulado de tal forma que se pueda usar libremente a escala masiva, como sucede con el Wi-Fi. En cambio, su ventaja principal con respecto a la localización en interiores usando Wi-Fi, es su mayor robustez ante cambios del entorno (puertas, paredes, presencia o movimientos de personas, etc.).

Sus posibilidades de futuro, sin embargo, no se encuentran realmente en el campo de la localización, sino que más bien pasan por ser un sustituto del USB para conseguir conexiones entre dispositivos sin cables y altas tasas de transferencia.

Existe una compañía llamada Ubisense que posee un sistema comercial que utiliza la tecnología UWB. El producto de localización combina dos técnicas de localización; una basada en la diferencia de los tiempos de llegada de las señales (TDOA) y otra basada en la medida de los ángulos de llegada (AOA). Estas técnicas serán explicadas en apartados posteriores.

Este sistema híbrido funciona con una alta precisión y tiene en su haber algunos casos de éxito curiosos sin comentar sus aplicaciones en logística. En Inglaterra este sistema se ha implantado en una planta de energía nuclear integrado con un detector de radiación y de esa forma realizan un seguimiento de los trabajadores estudiando los niveles de radiación a los que están expuestos. En ese escenario las celdas tienen unas dimensiones de 15 metros como máximo.

Otro caso de éxito relevante es la implantación de este sistema para el ejército norteamericano en una sede de la división de operaciones. Su utilidad consiste en

localizar a los soldados en las maniobras de entrenamiento para estudiar sus movimientos en simulacros de operaciones.

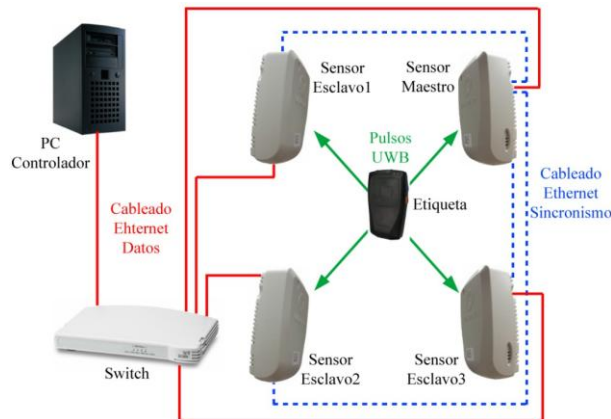


Figura 2.22: Arquitectura del sistema de Ubisense

2.2.9. Zig-Bee

Esta tecnología es la mejor en la actualidad en cuanto a la relación precisión-precio, junto con Wi-Fi, para la localización en interiores, y su viabilidad ya ha sido estudiada en algunos trabajos. Sin embargo, su alcance es más reducido aún que en Wi-Fi y la señal fluctúa de igual manera cuando cambian las condiciones de la planta, o con el movimiento de personas. Al igual que en Bluetooth las redes basadas en Zigbee hacen uso del indicador RSSI para realizar la estimación de la posición.

La ventaja principal de esta tecnología es su bajo coste, la baja potencia de emisión y su fácil integración, pero su bajo ancho de banda hace que su utilidad sea reducida. Zigbee está pensado para el ámbito de la domótica por lo que el posicionamiento es una aplicación de gran utilidad; sin embargo el despliegue de las redes Wi-Fi sumado al problema del ancho de banda hace posible creer que el posicionamiento con redes WiFi tenga más éxito que con Zig-Bee.

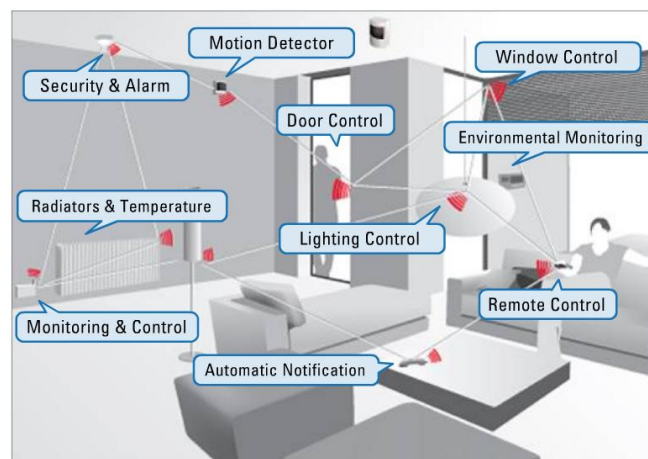


Figura 2.23: Sistema domótico con una red de sensores Zigbee

2.2.10. Comparativa

En la figura 2.24 se muestran la mayoría de las tecnologías explicadas anteriormente. La interpretación correcta de esta tabla pasa por entender la disposición de los rectángulos de cada tecnología. En el eje x se muestra la precisión del sistema y la anchura del rectángulo de cada tecnología muestra el rango de precisiones que se pueden conseguir. En el eje y se muestra el desarrollo tecnológico variando desde tecnologías en desarrollo hasta sistemas comerciales. El borde inferior de cada rectángulo muestra la posición actual de la tecnología y el borde superior la predicción estimada del alcance de la misma en los próximos años.

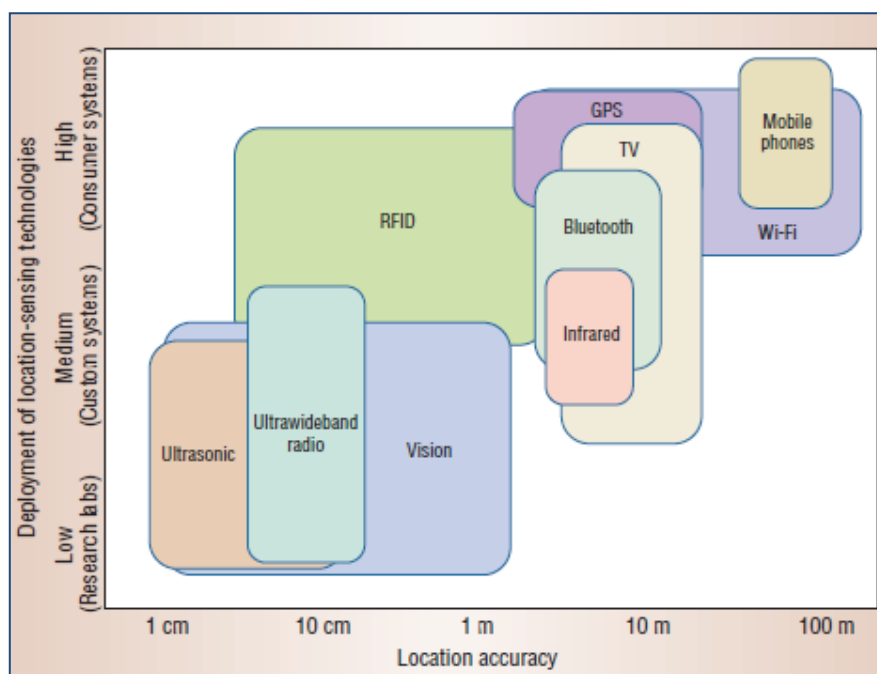


Figura 2.24: Comparativa de tecnologías

La información que extraemos de la figura anterior es un claro dominio de las redes WiFi y de la tecnología GPS en el mercado del consumidor a pesar de no ser los sistemas más precisos. UWB y RFID son muy precisas, pero tal y como hemos visto son sistemas caros y por ello su entrada al mercado comercial es difícil. Todas ellas tienen un ejemplo comercial, más o menos conocido, pero sólo unos pocos tienen un futuro prometedor.

La localización de telefonía móvil es un claro sistema en auge que provocará mejoras importantes en campos como la publicidad o los sistemas de emergencia. Aún no se consiguen altas precisiones pero debido a su importancia en un futuro, se está produciendo una alta inversión mundial en su desarrollo tal como se verá en el bloque 4.

Finalmente en la siguiente tabla se muestra un resumen de las tecnologías de localización indoor.

Tabla 2.4	WiFi	UWB	Zigbee	RFID	Bluetooth	Infrarrojos
Precisión	Rango de 1 a 5 metros según la complejidad del área	Hasta 15 cm con alta densidad de receptores	Según densidad de balizas pero como máximo 1 -2 metros	Hasta varios cm con alta densidad de etiquetas	Media con alto número de emisores	Media con un escenario adecuado
Despliegue y calibración	1. Poca infraestructura 2. Implementación rápida y barata 3. Calibración rápida salvo con mapa de potencias	1. Despliegue difícil por el sincronismo 2. Las computación está centralizada	1. Despliegue sencillo y económico 2. Computación centralizada o distribuida en función de los algoritmos elegidos	1. Gran infraestructura 2. Elevado precio	1. Despliegue barato con pocos emisores 2. Calibración muy sencilla	1. Despliegue caro 2. Número muy elevado de emisores
Precisión escalable	Incrementando Aps	Incrementando detectores	Incrementando balizas	Incrementando el número de etiquetas	Incrementando el número de emisores	Aumentando el número de emisores
Inmunidad frente a otros sistemas	Susceptible a interferencias	Fuerte frente a la distorsión multipath	Alta inmunidad en ambientes con baja SNR	Según la frecuencia elegida	Susceptible a interferencias	Alta
Potencialidades	Tecnología extendida y de bajo coste	Utiliza TDOA en vez de RSS mejorando los resultados	Tecnología barata Alta presencia en la domótica	Tecnología extendida	Consumo de energía bajo Coexistencia con otras redes bluetooth	Tecnología aplicada en robótica
Limitaciones	Pérdida de señal debida a obstáculos o altos niveles de ruido	Precio elevado y consumo de energía alto	Tiempo de computación elevado	Consumo de energía medio Utilización de RSS	Para aumentar rango de alcance gran consumo de energía	Corto alcance LoS

2.3. TÉCNICAS DE LOCALIZACIÓN

Hay diversas técnicas de localización que usan la tecnología 802.11. Según [1] se pueden clasificar en tres tipos diferentes tal y como muestra la figura 2.25. A continuación se va a describir cada una de ellas y se explicará cuál se ha elegido en este proyecto y por qué.

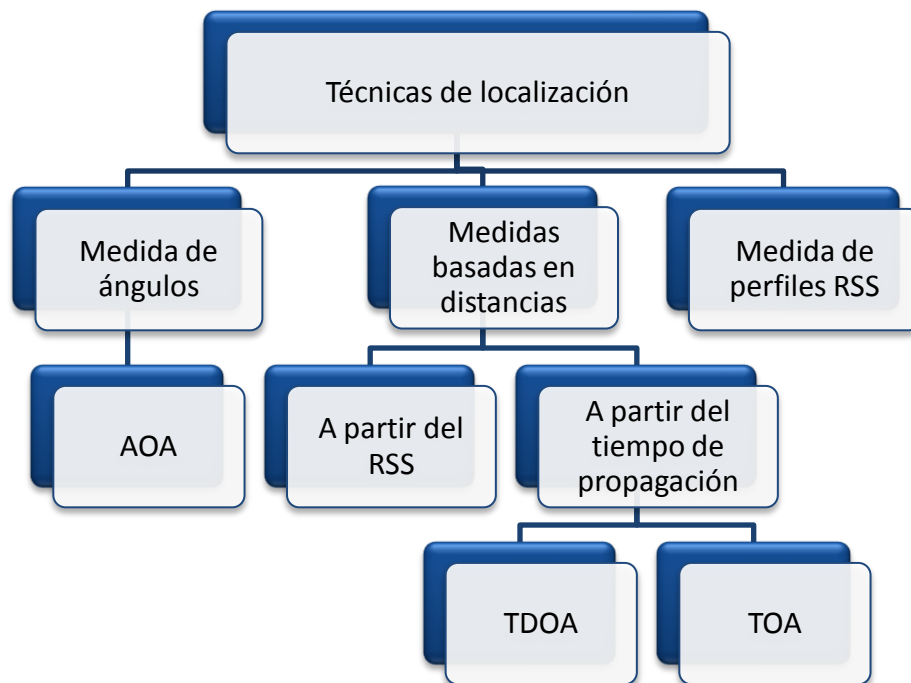


Figura 2.25: Clasificación de técnicas de localización

2.3.1. TOA

La técnica TOA (Time of Arrival – Tiempo de llegada) se basa en el hecho de que una señal electromagnética tarda un tiempo en llegar de emisor a receptor, tiempo de propagación. Conociendo la relación lineal que existe entre el tiempo de propagación y la distancia recorrida, se puede hallar la posición de un emisor utilizando varios receptores mediante técnicas de triangulación.

Para poder llevar a cabo las anteriores medidas, se requiere una sincronización perfecta entre el emisor y el receptor. Para conseguirla, harían falta relojes muy precisos y por lo tanto el sistema tendría un coste muy elevado.

Para hacer frente a este problema, una alternativa es la técnica TDOA (Time Difference of Arrival), que hace uso de dos señales, cada una con velocidades de propagación diferentes. La primera de ellas es una señal de radio que tiene un tiempo de propagación T_R ; la segunda, en cambio, es una señal acústica cuyo tiempo de propagación se expresa como T_S . En la figura 2.26 se muestra el cálculo de la distancia al receptor en TDOA.

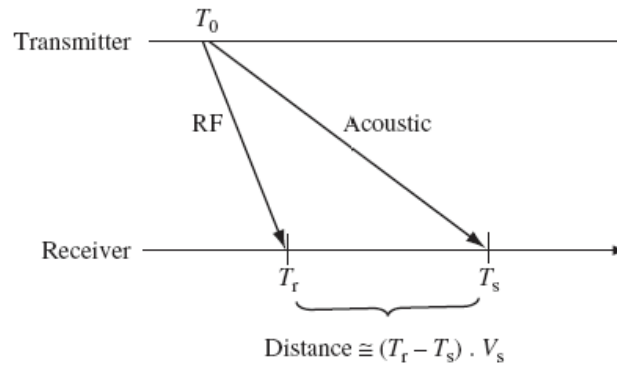


Figura 2.26: Cálculo de la distancia entre transmisor y receptor en TDOA.

Como la distancia que tarda en llegar una señal de radio de un transmisor a un receptor situado a pocos metros es muy pequeña, se puede asimilar a un tiempo 0. Por ello, observaremos cuánto tiempo más ha tardado la recepción de la señal acústica, y multiplicándolo por la velocidad de propagación de la señal acústica tendremos una medida bastante correcta de la distancia.

La ventaja de este tipo de TDOA, de combinar señales RF con ultrasonidos, respecto del TOA es que al medir primero el tiempo de forma relativa frente a la medida absoluta del TOA se puede disminuir el error porque se puede usar una resolución temporal mayor.

También se puede emplear otro tipo de TDOA, el estudiado a fondo en este proyecto, que compara por pares los tiempos de llegada de las señales de cada estación base (BTS) o sensor, mediante el uso de hipérbolas, las cuales cumplen la condición de que la distancia a cada BTS/sensor del par sea constante. Una representación gráfica de este método se muestra en la figura 2.27.

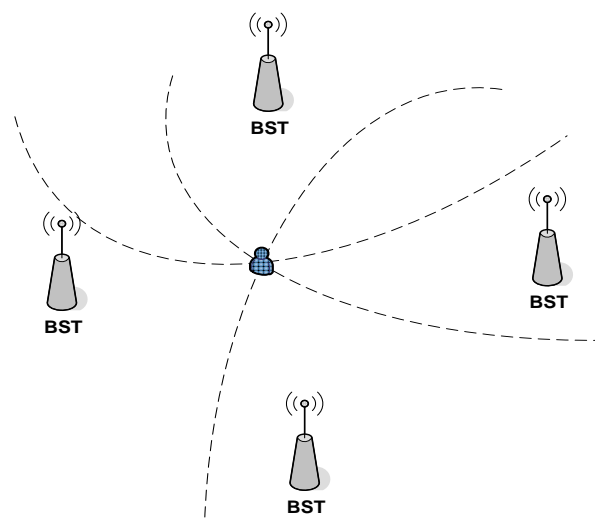


Figura 2.27: TDOA por comparación por pares de estaciones base

La técnica TOA/TDOA tiene los inconvenientes típicos de la localización en interiores, destacando el multitrayecto, provocando la llegada de varias señales con

retardos diferentes y por tanto las mediciones de distancias no son muy precisas. Por ello es necesaria la utilización de algoritmos para intentar paliar esos problemas y conseguir estimaciones precisas tal y como se verá en el bloque 3.

2.3.2. AoA

Las técnicas AoA (Angle of Arrival) son comunes en el procesamiento de señales con arrays. En lugar del uso de una sola antena, se despliegan varias antenas de forma que se pueda inferir el ángulo de llegada de la señal. Después de esto, la estimación de la localización se basa en la triangulación de estos ángulos. Aunque el sistema es bastante preciso, también es verdad que los dispositivos 802.11 no incluyen arrays de antenas, sino una, o como mucho dos, y por esta razón no se puede implementar un sistema de localización basado en AoA sin hardware adicional.

En realidad, la técnica del ángulo de llegada es efectiva sólo cuando no hay multitrayectos porque en caso de haberlos se puede pensar que la señal viene de otro lugar al medir el ángulo de una señal reflejada. Por ello, normalmente se necesita LoS para estimar el ángulo de un target hacia una antena emisora.

Al necesitarse LoS, las técnicas AoA son más adecuadas para entornos muy despejados como pueden ser los rurales, y son muy sensibles a variaciones de orientación de la antena emisora provocadas por el viento o tormentas, porque miden el ángulo de forma absoluta. La figura 2.28 muestra el cálculo de la localización con AoA.

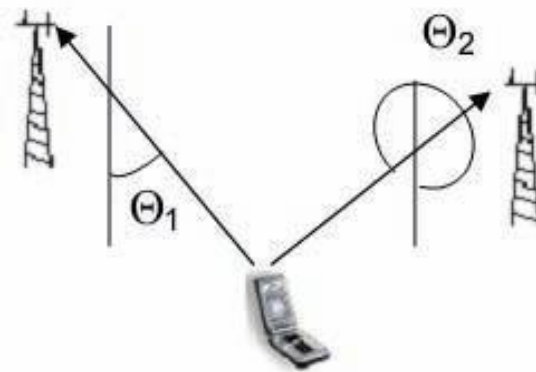


Figura 2.28: Sistema de localización por ángulo de llegada

2.3.3. RSS

Esta técnica usa la atenuación de la señal recibida para inferir la localización del terminal móvil. En base a esta atenuación se puede estimar la distancia que separa al Target de cada punto de acceso, pero la relación distancia / potencia de señal recibida no es muy clara al haber multitrayecto tal y como se ha explicado anteriormente. Sin embargo, ésta es la solución más simple conceptualmente y la más barata de las tres, al no necesitar ningún hardware adicional.

Además del inconveniente fundamental ya explicado, también hay desafíos adicionales, como en el caso de RSS por patrones, la construcción de un mapa radio lo más fiable como sea posible o la propia variabilidad de la RSS en función del uso de distintos tipos de antenas, fabricantes, o incluso del nivel de carga de la batería del dispositivo.

En el bloque 3 explicaremos un poco más en profundidad las diversas formas que existen de utilizar la medida del RSS para el posicionamiento.

2.3.4. Tabla comparativa

	AOA	TOA	TDOA	RSS (Patrones)
Requisitos	Array de antenas Rural: 2 BS Urbano: 3+ BS	Sincronismo 3+ estaciones	3+ estaciones	Entrenamiento previo (mapa de potencias)
Precisión	Media	Media	Alta	Media
Coste	Alto	Alto	Alto	Bajo
Cobertura	Alta	Alta	Alta	Baja
Problemas	NLOS Multitrayecto (evitable)	Multitrayecto Tiempo procesado	Antenas colineales Multitrayecto	Multipath fading Shadowing

Tabla 2.5: Comparación de técnicas de localización

Memoria

Sistemas y técnicas de localización

Bloque 3

3. Simulación de localización en entornos interiores

3.1. TIME DIFFERENCE OF ARRIVAL

El método Time Difference of Arrival (TDOA) utiliza relaciones geométricas basadas en diferencias de distancias entre el terminal a buscar o target y un número de sensores para determinar la posición del objetivo. El hecho de utilizar diferencias de distancias es una gran ventaja en el campo práctico ya que evita la necesidad de tener los terminales sincronizados. TDOA está basado en el principio de medida de distancias, Time-Of-Flight (TOF): la distancia entre un transmisor y un receptor es igual al tiempo de vuelo, o tiempo de propagación electromagnético, de la señal transmitida por la velocidad de propagación. Por lo tanto partiendo de los intervalos de tiempo medidos y utilizando la velocidad de propagación del medio se consigue obtener las distancias necesarias para la localización del target. TDOA finalmente localiza el target por medio de la intersección de hipérbolas o hiperboloides, según sean dos o tres dimensiones, con su foco centrado en el sensor de cada par.

La parte más importante de todo método de localización basado en tiempos es sin duda la obtención de los datos temporales. Para este propósito existen dos opciones dependiendo del tipo de sistema; sistemas unilaterales o multilaterales.

En el primer tipo de sistemas el target actúa como receptor; es decir, solamente recibe las señales transmitidas por los sensores. Asumiendo que el sistema sólo utiliza una frecuencia, es necesaria la utilización de un protocolo de multiplexación temporal. La utilización de técnicas de multiplexación es necesaria para obtener los datos temporales de cada sensor para que no se produzcan interferencias entre las señales. El escenario más común consiste en que los relojes de los transmisores estén sincronizados y los transmisores envíen tramas beacon periódicas (balizas) a tiempos escalonados para evitar solapamientos en las transmisiones.

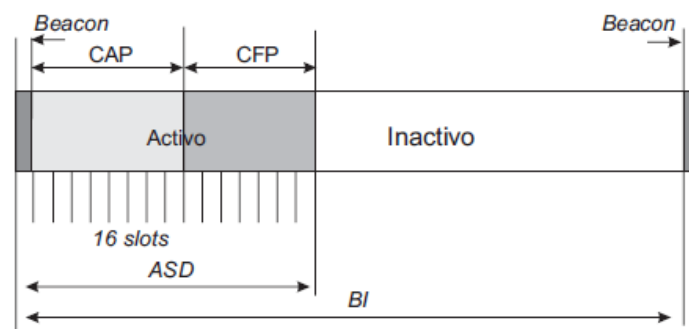


Figura 3.1: Supertrama Zigbee

El target conoce la diferencia de tiempo de las transmisiones de los epochs por parte de los transmisores, por lo que puede obtener el TDOA como el intervalo de tiempo entre los epochs de las señales recibidas, medidas con el reloj del receptor, menos la diferencia de tiempo entre las dos transmisiones. En la figura 3.2 se observa el diagrama de tiempos de un sistema unilateral con tres transmisores donde los TDOA obtenidos por el target serían los siguientes:

$$\Delta t_{2,1} = (t_2 - t_1) - \tau_2$$

$$\Delta t_{3,1} = (t_3 - t_1) - \tau_3$$

$$\Delta t_{3,2} = (t_3 - t_2) - (\tau_3 - \tau_2)$$

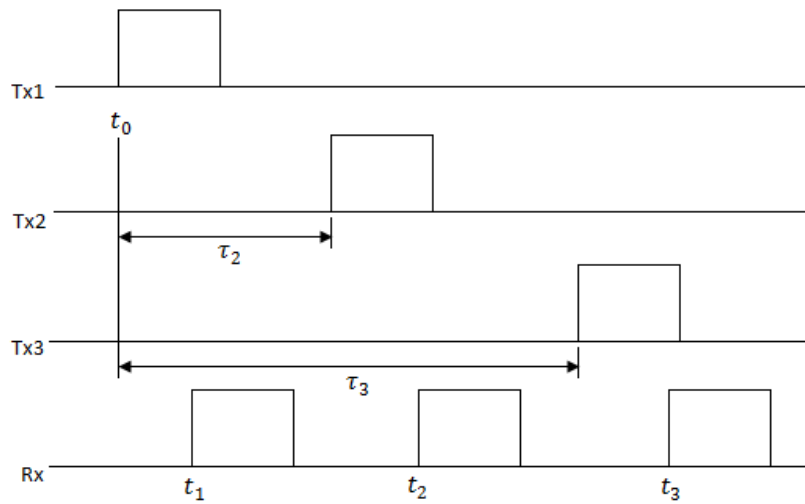


Figura 3.2: Diagrama de tiempos TDOA de un sistema unilateral

Un claro ejemplo de estos sistemas es el sistema de ayuda a la navegación Loran-C. Este sistema es utilizado en muchos países, entre ellos los Estados Unidos de América y Japón y funciona en frecuencias del espectro magnético entre 90 y 100 kHz. Actualmente está siendo abandonado por el sistema GPS

Por otro lado, en los sistemas multilaterales los cálculos de localización se realizan independientemente del target. Estos cálculos se realizan en los sensores o incluso en una infraestructura de red exterior. En este caso el target es el que transmite y los receptores monitorean simultáneamente la red. En la figura 3.3 tenemos un ejemplo de sistema multilateral TDOA con 3 sensores o receptores. Los relojes de los receptores deben estar sincronizados para obtener los TDOA de una manera más simple que en el caso anterior.

$$\Delta t_{2,1} = t_2 - t_1$$

$$\Delta t_{3,1} = t_3 - t_1$$

$$\Delta t_{3,2} = t_3 - t_2$$

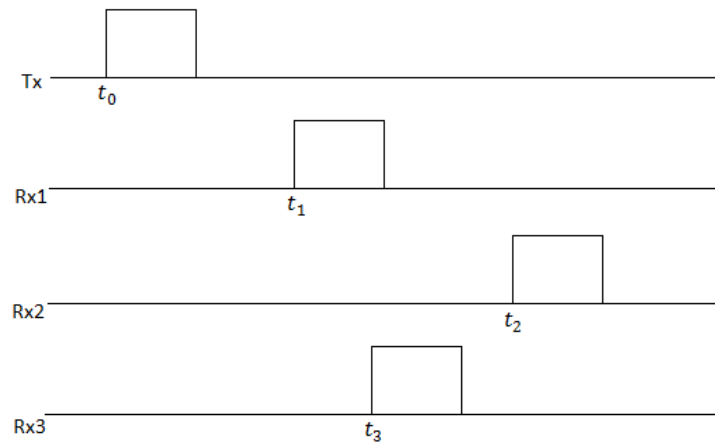


Figura 3.3: Diagrama de tiempos TDOA de un sistema multilateral

El sistema RADAR o la medida de distancias basada en la técnica OFDM son ejemplos de sistemas multilaterales en los que el target actúa como transmisor y los sensores del sistema actúan como receptores.

3.1.1. Descripción matemática

A la hora de conocer el método TDOA, primero es necesario entenderlo matemáticamente. Lo primero a tener en cuenta es que para realizar una localización 2D son necesarios por lo menos tres sensores. De este modo se tiene la intersección de dos hipérbolas aunque la posibilidad de obtener una ambigüedad en cuanto al resultado puede ser resuelto si se tienen 4 sensores o lo que es lo mismo tres hipérbolas. En el caso de localización 3D es necesaria la utilización de cuatro sensores; que equivalen a tres hiperboloides para intersecar y así obtener la componente z del target. En la figura 3.4 se muestra un ejemplo de localización TDOA con tres sensores válido para posicionamiento 2D.

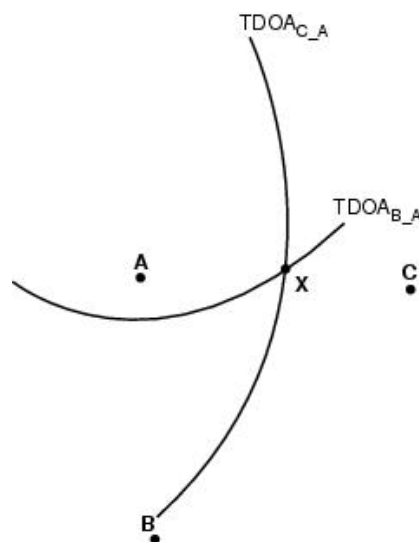


Figura 3.4: Localización TDOA en 2 dimensiones

A continuación se van a mostrar las ecuaciones de un sistema multilateral ideal con cuatro sensores con localizaciones conocidas en los que no se tienen en cuenta efectos como reflexiones, difracciones u otros elementos que alteren las medidas. Las variables del sistema son las siguientes:

- Terminal target con coordenadas (x, y, z)
- Sensores
 - Terminal i con coordenadas (x_i, y_i, z_i) con $i = 1, \dots, 4$
- Velocidad de propagación c

Los tiempos de vuelo entre el target y cada sensor serian simplemente las distancias entre cada par de terminales divididas por la velocidad de propagación.

$$T_1 = \frac{1}{c} R_1 = \frac{1}{c} \sqrt{(x - x_1)^2 + (y - y_1)^2 + (z - z_1)^2}$$

$$T_2 = \frac{1}{c} R_2 = \frac{1}{c} \sqrt{(x - x_2)^2 + (y - y_2)^2 + (z - z_2)^2}$$

$$T_3 = \frac{1}{c} R_3 = \frac{1}{c} \sqrt{(x - x_3)^2 + (y - y_3)^2 + (z - z_3)^2}$$

$$T_4 = \frac{1}{c} R_4 = \frac{1}{c} \sqrt{(x - x_4)^2 + (y - y_4)^2 + (z - z_4)^2}$$

Una vez obtenidos los tiempos de vuelo, el siguiente paso consiste en obtener las diferencias de tiempos de vuelo. Cada ecuación siguiente define un hiperboloide y la intersección de todos ellos corresponde a la localización del target.

$$\tau_{21} = \frac{1}{c} (R_2 - R_1) = T_2 - T_1$$

$$\tau_{31} = \frac{1}{c} (R_3 - R_1) = T_3 - T_1$$

$$\tau_{41} = \frac{1}{c} (R_4 - R_1) = T_4 - T_1$$

Si se desarrollan las ecuaciones se obtienen las siguientes diferencias de tiempos

$$\tau_{21} = \frac{1}{c} (\sqrt{(x - x_2)^2 + (y - y_2)^2 + (z - z_2)^2} - \sqrt{(x - x_1)^2 + (y - y_1)^2 + (z - z_1)^2})$$

$$\tau_{31} = \frac{1}{c} (\sqrt{(x - x_3)^2 + (y - y_3)^2 + (z - z_3)^2} - \sqrt{(x - x_1)^2 + (y - y_1)^2 + (z - z_1)^2})$$

$$\tau_{41} = \frac{1}{c} (\sqrt{(x - x_4)^2 + (y - y_4)^2 + (z - z_4)^2} - \sqrt{(x - x_1)^2 + (y - y_1)^2 + (z - z_1)^2})$$

Como se ha comentado, la solución gráfica del problema anterior consiste en hallar el punto de intersección de los tres hiperboloides, pero en una aplicación real no existe dicho punto. Esto es debido a la aparición de efectos, como el ruido, la reflexión o la difracción, que provocan errores en las medidas y como consecuencia, matemáticamente, aparece un término más en las ecuaciones.

$$T_i = \frac{1}{c} \sqrt{(x - x_i)^2 + (y - y_i)^2 + (z - z_i)^2} + \Delta_0$$

Siendo:

$$i = 1, \dots, 4$$

Δ_0 = término extra debido a imprecisiones

La resolución de ecuaciones no lineales hiperbólicas no es nada trivial por lo que es necesario acudir a técnicas de optimización avanzadas. Hallar una solución por tanto, supone acudir a la literatura donde se encuentran diferentes métodos [2].

Tras realizar una búsqueda exhaustiva se pueden diferenciar dos tipos de métodos según la disposición de los sensores,

1. sensores dispuestos linealmente; es decir en el mismo plano
2. sensores dispuestos arbitrariamente

Los primeros no son tan versátiles porque en un escenario real los sensores o estaciones base no suelen estar dispuestos a la misma altura. Lo normal es adaptarse según las necesidades que tengamos o según el propio diseño arquitectónico de nuestro escenario. De todos modos, en este tipo de métodos, como el método de Hahn o el método de Carter, la estimación del target es mucho más simple.

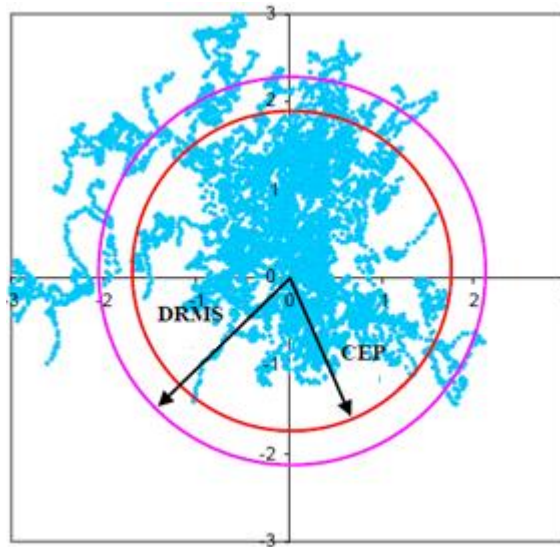
Por lo tanto el estudio se centrará en los segundos, de los cuales se pueden listar los más relevantes a continuación. Dentro de este estudio, se intenta mostrar las fortalezas y debilidades de cada uno y finalmente en los capítulos de resultados mostrar ejemplos prácticos de algunos de ellos.

- Método de la serie de Taylor
- Método de Fang
- Método de interpolación esférica
- Método de Chan
- Método de Friedlander
- Método de Levenberg – Marquardt
- Método de Divide y Conquista
- Método de Steepest Descent

Es necesario introducir en este punto, unos cuantos conceptos claves relacionados con la precisión de las estimaciones que nos proporcionan los anteriores métodos

aunque solo sea de carácter informativo. Dependiendo del número de dimensiones del escenario en el que nos encontremos tenemos que hablar de unos indicadores u otros.

En el caso de 2D, destacamos dos indicadores numéricos importantes. El primero es el CEP (Circular Error Probability) que indica el radio de un círculo alrededor de la posición estimada que contiene el 50% de de las estimaciones generadas y el segundo es el DRMS (Distance Root Mean Squared) que es la raíz cuadrada del promedio de errores cuadrados de las estimaciones frente a la posición real, y al igual que CEP también define un círculo. En el caso del CEP, si se quiere tener un círculo con probabilidad del 95% de estimaciones, es necesario acudir al indicador R95.



$$\text{CEP} = 0.62 * \sigma_y + 0.56 * \sigma_x$$

$$\text{DRMS} = \sqrt{\sigma_x^2 + \sigma_y^2}$$

Figura 3.5: Indicadores CEP y DRMS gráficamente

Siendo:

$$\sigma_x^2 = \text{Varianza de la coordenada x}$$

$$\sigma_y^2 = \text{Varianza de la coordenada y}$$

Si el interés se centra en 3D, como es el caso de este proyecto, se siguen teniendo los mismos indicadores pero aplicados a 3 dimensiones. Es decir, la definición de SEP (Spherical Error Probability) es idéntica al CEP pero aplicada a tres dimensiones; radio de una esfera en la que se concentran el 50% de las estimaciones. Por otro lado el RMSE (Mean Radial Spherical Error) que se aplica en tres dimensiones equivale a la definición de DRMS; raíz cuadrada del promedio de los errores cuadrados de las estimaciones que nos define una esfera.

$$\text{SEP} = 0.51 * (\sigma_x + \sigma_y + \sigma_z)$$

$$\text{MRSE} = \sqrt{\sigma_x^2 + \sigma_y^2 + \sigma_z^2}$$

El objetivo de este proyecto ha sido el estudio de las diferentes técnicas y realizar una comparación de las más importantes. En los siguientes apartados se va a realizar un estudio teórico de los diferentes métodos profundizando en algunos de ellos por su relevancia para el proyecto. Además al final se realizará una comparación teórica de todos ellos mostrando sus debilidades y sus fortalezas.

3.1.2. Algoritmos de resolución de ecuaciones no lineales hiperbólicas

3.1.2.1. Método de las series de Taylor

El método de las series de Taylor [3] fue expuesto por primera vez en 1976 por W. Foy y sirve para obtener una estimación precisa con niveles razonables de ruido. Es un método robusto que puede obtener resultados precisos y además es capaz de utilizar medidas redundantes para mejorar la estimación de la posición del target. El método de Taylor consiste en linealizar las ecuaciones formadas por las diferencias de tiempos obtenidas a partir de los tiempos de vuelo que se han visto anteriormente. Para ello, primero hace uso de la expansión en series de Taylor para después utilizar un método iterativo de resolución de sistemas de ecuaciones lineales.

El método iterativo comienza con una estimación inicial que va mejorando progresivamente determinando la solución local por mínimos cuadrados (Least Squares). El problema de este método es que requiere de una estimación inicial cercana a la solución final y por tanto en caso de no realizarla correctamente se puede producir una no convergencia de la solución.

A continuación se va a mostrar el procedimiento matemático de este método. Las variables iniciales del procedimiento son las estudiadas en el apartado anterior; es decir:

- Posición real del target (x, y, z)
- Posición de los sensores (x_i, y_i, z_i) con $i = 1, \dots, N$
- N número de sensores
- Estimación inicial $(\hat{x}, \hat{y}, \hat{z})$
- $m_{k,1}$ Diferencia de tiempos de llegada entre sensor k y sensor 1

El primer paso consiste en calcular las distancias entre los sensores y la estimación inicial.

$$r_i = \sqrt{(x_i - \hat{x})^2 + (y_i - \hat{y})^2 + (z_i - \hat{z})^2} \text{ siendo } i = 1, \dots, N$$

Seguidamente, se pasa a calcular la matriz A definida por Wade H. Foy de la siguiente forma:

$$A = \begin{bmatrix} \frac{x_1 - \hat{x}}{r_1} - \frac{x_2 - \hat{x}}{r_2} & \frac{y_1 - \hat{y}}{r_1} - \frac{y_2 - \hat{y}}{r_2} & \frac{z_1 - \hat{z}}{r_1} - \frac{z_2 - \hat{z}}{r_2} \\ \frac{x_1 - \hat{x}}{r_1} - \frac{x_3 - \hat{x}}{r_3} & \frac{y_1 - \hat{y}}{r_1} - \frac{y_3 - \hat{y}}{r_3} & \frac{z_1 - \hat{z}}{r_1} - \frac{z_3 - \hat{z}}{r_3} \\ \vdots & \vdots & \vdots \\ \frac{x_1 - \hat{x}}{r_1} - \frac{x_i - \hat{x}}{r_i} & \frac{y_1 - \hat{y}}{r_1} - \frac{y_i - \hat{y}}{r_i} & \frac{z_1 - \hat{z}}{r_1} - \frac{z_i - \hat{z}}{r_i} \end{bmatrix} \text{ siendo } i = 2, \dots, N$$

Por otro lado es necesario definir los valores de f_k , que son las diferencias de distancias, pero utilizando la estimación inicial, es decir:

$$f_k = r_k - r_1 \text{ siendo } k = 2, \dots, N$$

Se continua construyendo la matriz Z de la siguiente forma:

$$Z = \begin{bmatrix} m_{2,1} - f_2 \\ m_{3,1} - f_3 \\ \vdots \\ m_{i,1} - f_i \end{bmatrix} \text{ siendo } i = 2, \dots, N$$

Por último Foy toma los dos primeros términos de la expansión en series de Taylor de las ecuaciones de las diferencias de distancias y realiza la siguiente aproximación:

$$A * \delta \cong Z$$

Donde:

$$\delta = \begin{bmatrix} \delta_x \\ \delta_y \\ \delta_z \end{bmatrix} \text{ es la posición estimada}$$

Por lo tanto, despejando la solución estimada quedaría de la siguiente manera:

$$\delta = (A^T * A)^{-1} * A^T * Z$$

El método anterior es un proceso iterativo en el que la salida se aplica a la entrada cambiando la estimación inicial t repitiendo todos los cálculos explicados, es decir:

$$\begin{pmatrix} \hat{x} \\ \hat{y} \\ \hat{z} \end{pmatrix} = \begin{pmatrix} \hat{x} \\ \hat{y} \\ \hat{z} \end{pmatrix} + \begin{pmatrix} \delta_x \\ \delta_y \\ \delta_z \end{pmatrix}$$

Este proceso acaba cuando δ_x , δ_y y δ_z son esencialmente 0.

Tal como se ha comentado, la solución anterior es una aproximación y por lo tanto no se obtienen buenos resultados. La solución real pasaría por obtener la matriz de covarianzas de los tiempos de llegada, Q , ya que Taylor se aprovecha de las propiedades del ruido para mejorar sus estimaciones.

Para hacer uso de las propiedades del ruido, partiendo del vector de los tiempos de llegada se crea una matriz con esos tiempos de llegada a los que se añaden M vectores más con esos tiempos más un término de ruido, creado de forma aleatoria, de distribución normal de media 0 y varianza 1 ns para cada uno de ellos. Por ejemplo en el caso de 4 transmisores el procedimiento sería el siguiente:

$$\begin{bmatrix} \text{TOA}_1 & \text{TOA}_2 & \text{TOA}_3 & \text{TOA}_4 \\ \text{TOA}_1 + \sigma_n & \text{TOA}_2 + \sigma_n & \text{TOA}_3 + \sigma_n & \text{TOA}_4 + \sigma_n \\ \vdots & \vdots & \vdots & \vdots \\ \text{TOA}_1 + \sigma_n & \text{TOA}_2 + \sigma_n & \text{TOA}_3 + \sigma_n & \text{TOA}_4 + \sigma_n \end{bmatrix} (M \times 4)$$

Siendo σ_n el término de ruido aditivo y M el número de medidas que se deseen utilizar; por ejemplo 100.

A partir de esta matriz se calcula la matriz de diferencias de tiempos de llegada y por último la matriz de covarianzas Q . El proceso es el mismo que se ha explicado anteriormente pero en este caso el último cálculo queda de la siguiente forma:

$$\delta = (A^T * Q^{-1} * A)^{-1} * A^T * Q^{-1} * Z$$

Como conclusión, se observa que la posición inicial es una debilidad de este método por ser su fundamento clave; por lo que una variación de esta provoca grandes diferencias en las soluciones. Entre sus ventajas se pueden mencionar su baja carga computacional y la capacidad de aprovechar las medidas redundantes para el proceso.

3.1.2.2. Método de Fang

El método de Fang [4] como su propio nombre indica fue introducido por Bertrand T. Fang en 1990. Este método parte directamente de las ecuaciones anteriormente estudiadas pero linealizadas, para hallar una solución exacta:

$$R_{i1}^2 + 2 * R_{i1} * R_1 + R_1^2 = x_i^2 + y_i^2 + z_i^2 - 2 * x_i * x - 2 * y_i * y - 2 * z_i * z + x^2 + y^2 + z^2$$

$$\text{Siendo: } R_{i1} = R_i - R_1$$

$$i = 1, \dots, N$$

Este método tiene una limitación en cuanto al número de ecuaciones a resolver ya que tiene la restricción de tener el mismo número de ecuaciones que de incógnitas. Por

lo tanto, en un escenario con localización en tres dimensiones, se deben tener 3 ecuaciones TDOA o lo que es lo mismo un escenario con 4 sensores. La consecuencia directa de esta limitación es que este método no es capaz de aprovechar medidas redundantes para calcular la estimación de la posición del target.

Se va a describir el método matemáticamente para entenderlo un poco mejor y para ello se parte del escenario de inicio del cual parte el autor. En el mismo existen tres estaciones base, de las cuales una coincide con el origen de coordenadas del sistema, tal y como se muestra en la siguiente figura.

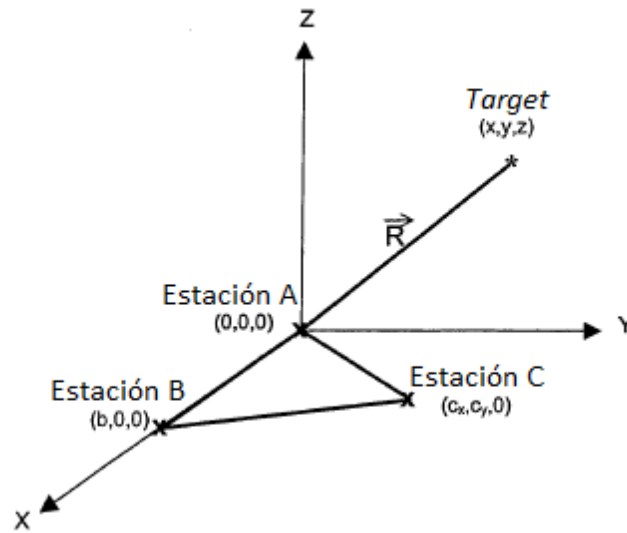


Figura 3.6: Escenario posicionamiento del método de Fang

Partiendo de los tiempos de llegada en las estaciones A, B y C, las diferencias de distancias entre el target y las estaciones se calcularían de la siguiente forma

$$R_{AB} = \sqrt{x^2 + y^2 + z^2} - \sqrt{(x - b)^2 + y^2 + z^2}$$

$$R_{AC} = \sqrt{x^2 + y^2 + z^2} - \sqrt{(x - c_x)^2 + (y - c_y)^2 + z^2}$$

Los valores de b , c_x y c_y son las coordenadas de las estaciones tal y como se ve en la figura 1.5. Si ahora se eleva al cuadrado y se simplifica

$$R_{AB}^2 - b^2 + 2 * b * x = 2 * R_{AB} * \sqrt{x^2 + y^2 + z^2}$$

$$R_{AC}^2 - c^2 + 2 * c_x * x + 2 * c_y * y = 2 * R_{AC} * \sqrt{x^2 + y^2 + z^2}$$

$$\text{siendo } c = \sqrt{c_x^2 + c_y^2}$$

Si se opera con las dos ecuaciones anteriores se obtiene una ecuación de la siguiente forma

$$y = g * x + h$$

donde

$$g = \frac{R_{AC} * \left(\frac{b}{R_{AB}}\right) - c_x}{c_y}$$

$$h = \frac{c^2 - R_{AC}^2 + R_{AC} * R_{AB} \left(1 - \left(\frac{b}{R_{AB}}\right)^2\right)}{2 * c_y}$$

Si se sustituye ahora en ecuaciones anteriores se obtiene lo siguiente

$$d * x^2 + e * x + f = z^2$$

Donde

$$d = -\left\{1 - \left(\frac{b}{R_{AB}}\right)^2 + g^2\right\}$$

$$e = b * \left\{1 - \left(\frac{b}{R_{AB}}\right)^2\right\} - 2 * g * h$$

$$f = \left(\frac{R_{AB}^2}{4}\right) * \left\{1 - \left(\frac{b}{R_{AB}}\right)^2\right\}^2 - h^2$$

Si ahora se utiliza otra estación base ya es posible resolver el sistema. De esta forma añadiendo la estación D con coordenadas $(d_x, d_y, 0)$ se obtiene la siguiente ecuación de diferencia de distancias

$$R_{AD} = \sqrt{x^2 + y^2 + z^2} - \sqrt{(x - d_x)^2 + (y - d_y)^2 + z^2}$$

Si se sigue el mismo proceso que se ha utilizado entre las estaciones A, B y C se consigue la segunda ecuación necesaria para resolver el sistema

$$y = g' * x + h'$$

Siendo

$$g' = \frac{R_{AD} * \left(\frac{b}{R_{AB}}\right) - d_x}{d_y}$$

$$h' = \frac{d^2 - R_{AD}^2 + R_{AD} * R_{AB} \left(1 - \left(\frac{b}{R_{AB}}\right)^2\right)}{2 * d_y}$$

$$d = \sqrt{d_x^2 + d_y^2}$$

Y por último operando se obtiene

$$d' * x^2 + e' * x + f' = z^2$$

Siendo

$$d' = -\left\{1 - \left(\frac{b}{R_{AB}}\right)^2 + g'^2\right\}$$

$$e' = b * \left\{1 - \left(\frac{b}{R_{AB}}\right)^2\right\} - 2 * g' * h'$$

$$f' = \left(\frac{R_{AB}^2}{4}\right) * \left\{1 - \left(\frac{b}{R_{AB}}\right)^2\right\}^2 - h'^2$$

De esta forma si se restan las dos ecuaciones

$$(d - d') * x^2 + (e - e') * x + (f - f') = 0$$

Y se despeja la coordenada x

$$x = \frac{-(e - e') \pm \sqrt{(e - e')^2 - 4 * (d - d') * (f - f')}}{2 * (d - d')}$$

Para hallar las coordenadas y y z sólo es necesario sustituir en ecuaciones anteriores.

Por lo tanto siguiendo este procedimiento se puede obtener una solución para un escenario con 4 estaciones base o sensores pero como ya se ha comentado anteriormente no tiene la capacidad de utilizar la ayuda de estaciones redundantes. Tal como se ha visto, este método tiene unos cuantos requisitos en cuanto a la posición de las estaciones y además no es capaz de hacer uso medidas redundantes para mejorar sus estimaciones; por todo ello este método no va a ser incluido en las comparaciones del capítulo de resultados.

3.1.2.3. Método de interpolación esférica

El método de interpolación esférica fue introducido por Julius O. Smith y Jonathan S. Abel en 1987 [5]. Primero soluciona (x,y,z) en términos de R_1 , luego introduce el resultado intermedio en las ecuaciones linealizadas mostradas en el apartado anterior, despreciando R_1 por independencia, para generar ecuaciones cuya incógnita ahora es R_1 . Sustituyendo los valores de R_1 , que minimizan la ecuación error por mínimos cuadrados con el resultado intermedio, se obtiene el resultado final.

Una desventaja clara de este método, al contrario que el método de Fang por ejemplo, es su incapacidad de resolver el problema en el que el número de incógnitas es igual al número de ecuaciones basadas en estimaciones TDOA. Por lo tanto en la localización 3D, esta desventaja limita su utilización para escenarios con un número de transmisores superior a 4. La aparición de este método fue por una razón clara; mejorar en un orden de magnitud la inmunidad ante el ruido del método de intersección esférica.

El método de intersección esférica [6] tiene un funcionamiento similar al de interpolación. En ese caso también provee una solución en términos de R_1 , y posteriormente para hallar el valor de este término realiza una estimación por mínimos cuadrados de la ecuación de distancias entre sensores y target en vez de acudir directamente las ecuaciones linealizadas.

El método de interpolación esférica es más versátil pero como se ha visto, al considerar a R_1 independiente de (x,y,z) , la solución obtenida es subóptima. A continuación se va a explicar el proceso matemáticamente, cambiando la notación ligeramente para seguir el desarrollo de sus autores, además de explicar la razón por la que este método no funciona cuando el número de ecuaciones es igual al número de ecuaciones.

Se considera un escenario con N sensores, pero sin ninguna limitación en cuanto a sus coordenadas como pasaba en el método de Fang. Primero se explicará la notación empleada por Smith y Abel para seguir con el desarrollo matemático.

Es necesario tener un sensor como referencia, como en casi todos los métodos vistos y por ver, que emplean TDOA, a partir del cual se obtienen el resto de las medidas, en este caso se tomará por defecto el sensor número 1.

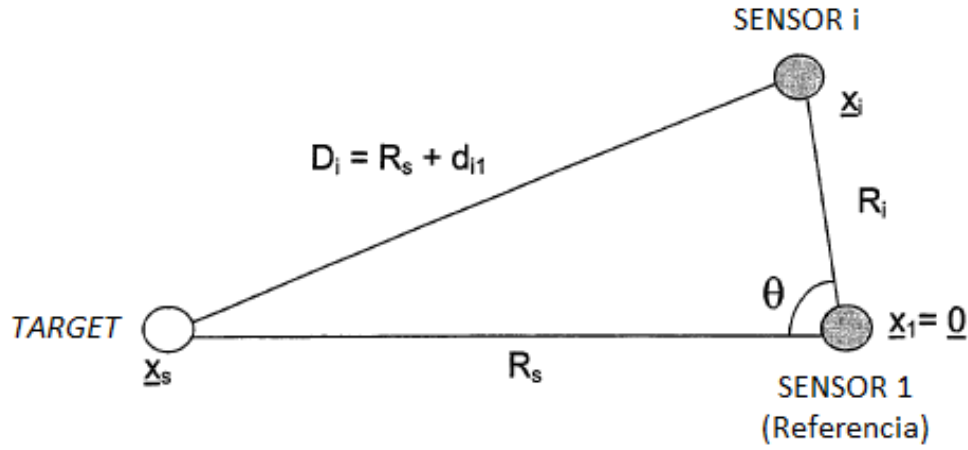


Figura 3.7: Gráfico de notación del método de interpolación esférica

En la figura 3.7 se observan las relaciones que existen entre la posición del target, un sensor cualquiera y el sensor de referencia. De esta figura se puede ver que la distancia entre un sensor cualquiera i y el target (D_i) es igual a la distancia entre el sensor de referencia y el target (R_s) más la diferencia de distancias entre el sensor i y el sensor de referencia (d_{i1}).

Si se utiliza el teorema del coseno para la ecuación de D_i se obtiene,

$$D_i^2 = (R_s + d_{i1})^2 = R_i^2 + R_s^2 - 2 * |\bar{x}_i| * |\bar{x}_s| * \cos \theta$$

Por otro lado, θ es el ángulo entre los vectores \bar{x}_i y \bar{x}_s y puede ser definido de la siguiente forma

$$\cos \theta = \frac{\bar{x}_i^T * \bar{x}_s}{|\bar{x}_i| * |\bar{x}_s|}$$

Por lo tanto sustituyendo en la primera ecuación,

$$(R_s + d_{i1})^2 = R_i^2 + R_s^2 - 2 * \bar{x}_i^T * \bar{x}_s$$

Si se opera y se simplifica,

$$\varepsilon_i = R_i^2 - d_{i1}^2 - 2 * R_s * d_{i1} - 2 * \bar{x}_i^T * \bar{x}_s$$

El término ε_i añadido refleja los errores introducidos en la ecuación debidos a imprecisiones en las medidas y será minimizado por mínimos cuadrados para obtener una estimación de la posición del target.

Si se consideran $N-1$ diferencias de distancias, la ecuación anterior se puede mostrar en forma de matrices de la siguiente forma,

$$\varepsilon = \delta - 2 * R_s * d - 2 * S * \bar{x}_s$$

Siendo

$$\delta = \begin{bmatrix} R_2^2 - d_{21}^2 \\ R_3^2 - d_{31}^2 \\ \vdots \\ R_N^2 - d_{N1}^2 \end{bmatrix} \quad d = \begin{bmatrix} d_{21} \\ d_{31} \\ \vdots \\ d_{N1} \end{bmatrix} \quad S = \begin{bmatrix} x_2 & y_2 & z_2 \\ x_3 & y_3 & z_3 \\ \vdots & \vdots & \vdots \\ x_N & y_N & z_N \end{bmatrix}$$

La matriz S representa las coordenadas de los sensores tras trasladar el sensor de referencia al origen de coordenadas. La solución por mínimos cuadrados de la ecuación anterior en función de \bar{x}_s sería la siguiente

$$\bar{x}_s = \frac{1}{2} * S_w^* * (\delta - 2 * R_s * d)$$

El signo $*$ representa conjugación compleja, es decir,

$$S_w^* = (S^T * S)^{-1} * S^T$$

y con esta ecuación se alcanza la minimización de $\varepsilon^T * \varepsilon$.

$$\boxed{\bar{x}_s = \frac{1}{2} * (S^T * S)^{-1} * S^T * (\delta - 2 * R_s * d)}$$

En este momento se realiza un inciso en la explicación matemática para mostrar una opción adicional muy interesante. Esta opción es la posibilidad de asignar pesos ponderados a las diferencias de distancias según la confianza que se tenga en cada una de ellas. Es decir, si se tienen unas medidas de sensores que se piensa que pueden tener altos niveles de ruido o términos altos de multitrayecto, se les puede asignar poco peso a la hora de calcular la posición estimada para de esta forma tener menor error en la estimación.

El rango de ponderación oscila entre un valor 0, para medidas que no se quieran utilizar, y un valor 1 para las medidas con alta fiabilidad. Utilizando esta opción alternativa, la ecuación a minimizar sería $\varepsilon^T * W * \varepsilon$ y esta solución sería,

$$S_w^* = (S^T * W * S)^{-1} * S^T * W$$

Tras el inciso, se va a continuar con el desarrollo matemático para descubrir la razón de la limitación en cuanto a sensores de este método. Si se sustituye la solución \bar{x}_s vista en la ecuación de ε para obtener una ecuación lineal en función de R_s ,

$$\varepsilon' = \delta - 2 * R_s * d - S * S_w^* * (\delta - 2 * R_s * d)$$

Simplificando,

$$\varepsilon' = (I - S * S_w^*) * (\delta - 2 * R_s * d)$$

En la ecuación anterior I es la matriz identidad y es necesario mencionar un término importante que es $S * S_w^*$. En el caso de tener cuatro sensores, al realizar el cálculo de dicho término se obtiene

$$S * S_w^* = I$$

Y por lo tanto,

$$\underline{\varepsilon'} = 0$$

Lo que significa que no existe estimación posible. Esta es la razón por la que son necesarios al menos 5 sensores para conseguir una posición estimada.

Ahora se va a mostrar cuál sería la solución con la interpolación esférica pero incluyendo la opción de la ponderación en las diferencias de tiempos. Si se cumple con el requisito del número de sensores y se añaden ponderaciones se obtiene,

$$\varepsilon'^T * W * \varepsilon' = (\delta - 2 * R_s * d)^T * (I - S * S_w^*) * W * (I - S * S_w^*) * (\delta - 2 * R_s * d)$$

El valor que minimiza la ecuación anterior por mínimos cuadrados es el siguiente

$$\widetilde{R_s} = \frac{d^T * (I - S * S_w^*) * W * (I - S * S_w^*) * \delta}{2 * d^T * (I - S * S_w^*) * W * (I - S * S_w^*) * d}$$

Y por último sustituyendo para obtener una estimación de la posición,

$$\bar{x}_s = \frac{1}{2} * S_w^* * (\delta - 2 * R_s * d)$$

$$\boxed{\mathbf{x}_s = \frac{1}{2} * (\mathbf{S}^T * \mathbf{W} * \mathbf{S})^{-1} * \mathbf{S} * \mathbf{W} * (\boldsymbol{\delta} - 2 * \mathbf{R}_s * \mathbf{d})}$$

3.1.2.4. Método de Chan

Y. T. Chan propuso en 1994 un método no iterativo para estimar la posición de un problema hiperbólico que alcanza resultados óptimos [7-8]. Cuando los errores en las medidas TDOA son pequeños, este método es una aproximación al estimador de máxima verosimilitud.

Al igual que Taylor, el método de Chan puede hacer uso de medidas redundantes para mejorar la solución pero necesita información a priori como el método de Fang. En cuanto a resultados, los estudios dicen que es ligeramente mejor que el método de interpolación esférica y que posee mejor tolerancia al ruido que el método de Divide-and-Conquer.

Tal y como se comentó respecto a los dos tipos de algoritmos que existen, sensores linealmente distribuidos y sensores aleatoriamente distribuidos, el método de

Chan trata ambas opciones. Sin embargo solamente se va a considerar la solución para la segunda opción porque es una solución más completa.

El método de Chan fue ideado principalmente para escenarios 2D y se diferenciaban dos situaciones, escenario con 3 sensores y escenario con más de 3 sensores. Se va a empezar explicando desde aquí para luego escalar a 3D tal y como Chan hizo.

Antes de empezar se va a recordar la notación vista con anterioridad y que se va a utilizar a continuación.

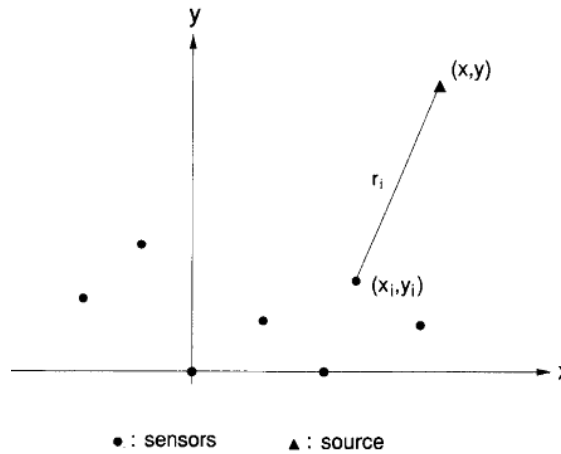


Figura 3.8: Localización en un escenario 2D

Tal y como se ve en la figura 3.8 la distancia entre el target y el sensor i es igual a,

$$\begin{aligned} r_i^2 &= (x_i - x)^2 + (y_i - y)^2 \\ &= K_i - 2 * x_i * x - 2 * y_i * y + x^2 + y^2 \end{aligned}$$

Siendo $K_i = x_i^2 + y_i^2$

Si se comienza con un escenario con 3 sensores, x e y pueden ser resueltos en función de r_1 de la siguiente forma,

$$\begin{bmatrix} x \\ y \end{bmatrix} = - \begin{bmatrix} x_{2,1} & y_{2,1} \\ x_{3,1} & y_{3,1} \end{bmatrix}^{-1} * \left\{ \begin{bmatrix} r_{2,1} \\ r_{3,1} \end{bmatrix} * r_1 + \frac{1}{2} \begin{bmatrix} r_{2,1}^2 - K_2 + K_1 \\ r_{3,1}^2 - K_3 + K_1 \end{bmatrix} \right\}$$

A continuación, el resultado es introducido en la ecuación vista arriba de r_i con $i = 1$. La raíz de la ecuación cuadrática obtenida se vuelve a introducir en la ecuación matricial para hallar la solución final. En caso de tener dos raíces positivas la ambigüedad de la solución debe ser resuelta restringiendo el transmisor a la región de interés.

En caso de que se tengan más de tres sensores ($N \geq 4$), el sistema está sobredimensionado ya que se poseen más ecuaciones que incógnitas. En este caso, se considera el vector $z = [x \ y \ R_1]$ como desconocido.

Lo primero es asumir que los términos de z no tienen relación entre sí para poder ser resueltos por mínimos cuadrados.

$$\begin{aligned} z &= \arg \min \{ (h - G_a * z)^T * \psi^{-1} * (h - G_a * z) \} \\ &= (G_a^T * \psi^{-1} * G_a)^{-1} * G_a^T * \psi^{-1} * h \end{aligned}$$

Siendo

$$\begin{aligned} h &= \frac{1}{2} \begin{bmatrix} r_{2,1}^2 - K_2 + K_1 \\ r_{3,1}^2 - K_3 + K_1 \\ \vdots \\ r_{N,1}^2 - K_N + K_1 \end{bmatrix} \\ G_a &= - \begin{bmatrix} x_{2,1} & y_{2,1} & r_{2,1} \\ x_{3,1} & y_{3,1} & r_{3,1} \\ \vdots & \vdots & \vdots \\ x_{N,1} & y_{N,1} & r_{N,1} \end{bmatrix} \\ \psi &= c^2 * B * Q * B \end{aligned}$$

En la última matriz es necesario explicar un par de matrices que no se han tenido en cuenta hasta ahora. Q es la covarianza del vector de TDOAs estimados $d = [d_{2,1}, \dots, d_{N,1}]$ y B contiene las distancias reales entre los sensores y el target, es decir las distancias sin ningún tipo de ruido. Tras esta explicación, es lógico pensar que de la ecuación expuesta ψ es desconocido, por lo que es necesaria una estimación.

Si el target está lejos de los sensores, la ecuación anterior puede aproximarse de la siguiente forma,

$$z \approx (G_a^T * Q^{-1} * G_a)^{-1} * G_a^T * Q^{-1} * h$$

Sin embargo, si el target está cerca, el procedimiento a seguir sería utilizar la ecuación aproximada para estimar la matriz B y luego introducir dicho valor en la ecuación completa sin aproximaciones.

La solución de este método requiere conocer la matriz de covarianza Q que en teoría no es conocida. En este momento aparecen dos opciones; la primera es utilizar el método de trabajo expuesto en el método de las series de Taylor en el que se añadía ruido de distribución normal a las medidas de tiempos de llegada y la segunda es estimar dicha matriz. Para dicha estimación, si las densidades espectrales de potencia de

ruido son similares en los sensores, esta matriz se puede reemplazar por una matriz con unos en la diagonal y 0.5 en el resto de elementos.

Visto el método 2D, lo siguiente es realizar un escalado a partir del conocimiento que se tiene para obtener las soluciones para escenarios 3D, que son los que realmente interesan para este proyecto. A igual que en el caso 2D ahora también aparecen dos casos de estudio en función del número de sensores que tenemos. El primero de ellos se produce cuando el número de sensores es igual a 4 produciendo 3 ecuaciones y el segundo cuando el número es superior a 4 y por tanto el sistema está sobredimensionado.

En el primer caso, $N=4$, la solución sería de la misma forma a la vista en dos dimensiones para tres sensores, pero en este caso se va a explicar más detenidamente. Las tres matrices a tener en cuenta son las siguientes;

$$A = - \begin{bmatrix} x_{2,1} & y_{2,1} & z_{2,1} \\ x_{3,1} & y_{3,1} & z_{3,1} \\ x_{4,1} & y_{4,1} & z_{4,1} \end{bmatrix}^{-1} \quad B = \frac{1}{2} \begin{bmatrix} r_{2,1}^2 - K_2 + K_1 \\ r_{3,1}^2 - K_3 + K_1 \\ r_{4,1}^2 - K_4 + K_1 \end{bmatrix} \quad R = \begin{bmatrix} r_{2,1} \\ r_{3,1} \\ r_{4,1} \end{bmatrix}$$

Al igual que antes hay que obtener una solución en función de r_1 para luego hallar la solución final. El procedimiento sería el siguiente:

$$E = A * R = - \begin{bmatrix} x_{2,1} & y_{2,1} & z_{2,1} \\ x_{3,1} & y_{3,1} & z_{3,1} \\ x_{4,1} & y_{4,1} & z_{4,1} \end{bmatrix}^{-1} * \begin{bmatrix} r_{2,1} \\ r_{3,1} \\ r_{4,1} \end{bmatrix}$$

$$F = A * B = - \begin{bmatrix} x_{2,1} & y_{2,1} & z_{2,1} \\ x_{3,1} & y_{3,1} & z_{3,1} \\ x_{4,1} & y_{4,1} & z_{4,1} \end{bmatrix}^{-1} * \frac{1}{2} \begin{bmatrix} r_{2,1}^2 - K_2 + K_1 \\ r_{3,1}^2 - K_3 + K_1 \\ r_{4,1}^2 - K_4 + K_1 \end{bmatrix}$$

Ahora ya se puede calcular los valores de la ecuación cuadrática $a * r_1^2 + b * r_1 + c = 0$

$$a = 1 - E' * E$$

$$b = 2 * [x_1 \quad y_1 \quad z_1] * E - F' * E$$

$$c = 2 * [x_1 \quad y_1 \quad z_1] * F - F' * F - K_1$$

Por lo tanto, se calcula la raíz

$$r_1 = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

Y la solución final sería,

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \mathbf{A} * (\mathbf{R} * \mathbf{r}_1 + \mathbf{B})$$

Ahora ya sólo queda la situación en la que existe redundancia en las medidas, es decir el número de sensores es superior a cuatro ($N \geq 5$). En este caso el vector desconocido es igual a $[x, y, z, r_1]$ y también se utiliza la aproximación vista en el caso de 2D con las matrices correspondientes,

$$\mathbf{z} \approx (\mathbf{G}_a^T * \mathbf{Q}^{-1} * \mathbf{G}_a)^{-1} * \mathbf{G}_a^T * \mathbf{Q}^{-1} * \mathbf{h}$$

$$\mathbf{G}_a = \begin{bmatrix} x_{2,1} & y_{2,1} & z_{2,1} & r_{2,1} \\ x_{3,1} & y_{3,1} & z_{3,1} & r_{3,1} \\ \vdots & \vdots & \vdots & \vdots \\ x_{N,1} & y_{N,1} & z_{N,1} & r_{N,1} \end{bmatrix}$$

$$\mathbf{h} = \frac{1}{2} \begin{bmatrix} r_{2,1}^2 - K_2 + K_1 \\ r_{3,1}^2 - K_3 + K_1 \\ \vdots \\ r_{N,1}^2 - K_N + K_1 \end{bmatrix}$$

$$\mathbf{Q} = \begin{bmatrix} 1 & 0.5 & 0.5 & \dots & 0.5 \\ 0.5 & 1 & 0.5 & \dots & 0.5 \\ \dots & \dots & \dots & \dots & \dots \\ 0.5 & 0.5 & 0.5 & \dots & 1 \end{bmatrix}$$

Siendo en este caso $K_i = x_i^2 + y_i^2 + z_i^2$ y \mathbf{Q} la matriz explicada anteriormente.

La primera aproximación se puede obtener de la siguiente forma

$$\begin{bmatrix} x \\ y \\ z \\ r_1 \end{bmatrix} = (\mathbf{G}_a^T * \mathbf{Q}^{-1} * \mathbf{G}_a)^{-1} * \mathbf{G}_a^T * \mathbf{Q}^{-1} * \mathbf{h}$$

y utilizando la solución obtenida de la variable r_1 se volvería a sustituir en la siguiente ecuación, al igual que en el caso de 2D, para alcanzar la solución final

$$\begin{bmatrix} x \\ y \\ z \\ r_1 \end{bmatrix} = (\mathbf{G}_a^T * \mathbf{Y}^{-1} * \mathbf{G}_a)^{-1} * \mathbf{G}_a^T * \mathbf{Y}^{-1} * \mathbf{h}$$

Siendo

$$\mathbf{Y} = \mathbf{c}^2 * \mathbf{B} * \mathbf{Q} * \mathbf{B}$$

$$B = \begin{bmatrix} r_2 & & & \\ & r_3 & & \\ & & \ddots & \\ & & & r_N \end{bmatrix}$$

3.1.2.5. Método de Friedlander

El método de Friedlander [9], al igual que la mayoría de los anteriores, lleva el nombre de su autor; Benjamin Friedlander. Este método de 1987 puede utilizar dos criterios de estimación para hallar la posición del target; mínimos cuadrados (LS) y mínimos cuadrados ponderados (Weighted LS). Al igual que con el método de interpolación esférica la opción de realizar ponderaciones es una mejora en la situación que se conozca la fidelidad de las medidas y no supone una complicación importante en los cálculos.

Friedlander comprobó que en un escenario con 4 estaciones base las soluciones obtenidas con ambos estimadores son idénticas. Sin embargo, si ese número crecía, el estimador WLS supera al estimador LS. Este método también utiliza las ecuaciones linealizadas mostradas anteriormente y asume que R_1 es independiente de la posición real del target y así prescinde de ese término, al igual que el método de interpolación esférica. Este método reduce la complejidad computacional de otros pero la solución alcanzada es subóptima porque elimina una relación fundamental en las ecuaciones.

La notación empleada por Friedlander varía un poco con la vista hasta ahora pero se parte de la misma ecuación linealizada que se expuso al comienzo de la explicación de método de Fang. Los elementos que cambian de notación son los siguientes: la distancia entre el origen de coordenadas y el sensor i es igual a R_{i0} y la distancia entre el target y el origen es igual a R_{s0} . En el estudio se parte de N sensores y se denomina al sensor j como el sensor de referencia.

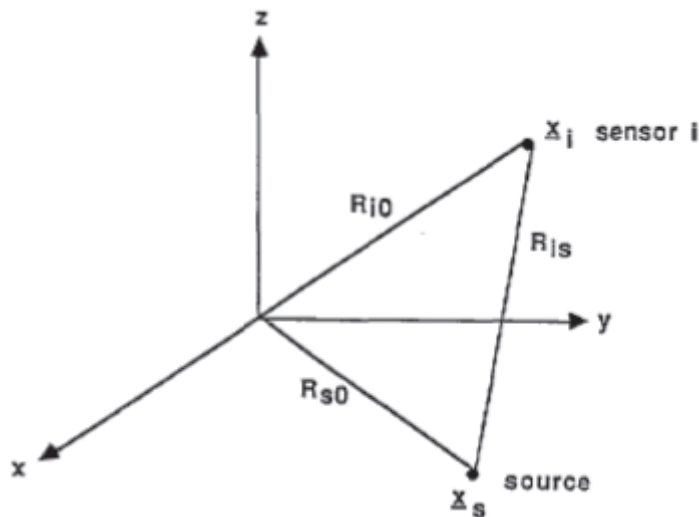


Figura 3.9: Notación del método de Friedlander

La ecuación linealizada quedaría de la siguiente forma

$$2 * x_i^T * x_s = R_{i0}^2 - r_{ij}^2 - 2 * R_{js} * r_{ij} + R_{s0}^2 - R_{js}^2$$

Si se realiza $i = j$ y $r_{jj} = 0$ se obtiene

$$2 * x_j^T * x_s = R_{j0}^2 + R_{s0}^2 - R_{js}^2$$

Restando las ecuaciones anteriores se obtiene

$$2 * (x_i - x_j)^T * x_s = (R_{i0}^2 - R_{j0}^2) - r_{ij}^2 - 2 * R_{js} * r_{ij}$$

Escribiendo la ecuación anterior en forma de matriz

$$S_j * x_s = \mu_j - R_{js} * \rho_j$$

Donde

$$S_j = \begin{bmatrix} (x_i - x_j) & (y_i - y_j) & (z_i - z_j) \\ \vdots & \vdots & \vdots \\ (x_{j-1} - x_j) & (y_{j-1} - y_j) & (z_{j-1} - z_j) \\ (x_{j+1} - x_j) & (y_{j+1} - y_j) & (z_{j+1} - z_j) \\ \vdots & \vdots & \vdots \\ (x_N - x_j) & (y_N - y_j) & (z_N - z_j) \end{bmatrix}$$

$$\mu_j = \frac{1}{2} * \begin{bmatrix} R_{i0}^2 - R_{j0}^2 - r_{ij}^2 \\ \vdots \\ R_{j-1,0}^2 - R_{j0}^2 - r_{j-1,j}^2 \\ R_{j+1,0}^2 - R_{j0}^2 - r_{j+1,j}^2 \\ \vdots \\ R_{N0}^2 - R_{j0}^2 - r_{Nj}^2 \end{bmatrix}$$

$$\rho_j = \begin{bmatrix} r_{1j} \\ \vdots \\ r_{j-1,j} \\ r_{j+1,j} \\ \vdots \\ r_{Nj} \end{bmatrix}$$

En este momento aparecen dos opciones dependiendo de si el término R_{js} , que muestra la distancia entre el sensor de referencia y el target, es conocido o no. En el caso de que se conozca la solución es inmediata. La solución estimada por mínimos cuadrados sería

$$\boxed{x_s = (S_j^T * S_j)^{-1} * S_j^T * (\mu_j - R_{js} * \rho_j)}$$

En el segundo caso, la solución es más complicada. En ese caso es necesario utilizar una matriz auxiliar que al multiplicar por ρ_j el resultado es nulo y así hacer desaparecer el término R_{js} desconocido. La matriz extra es

$$M_j = (I - Z) * D_j$$

Y está formada por

$$I = \begin{bmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & 1 \end{bmatrix} \text{ (matriz identidad)}$$

$$Z = \begin{bmatrix} 0 & 1 & \dots & 0 \\ \dots & \dots & \dots & \dots \\ \dots & 0 & \dots & 1 \\ 1 & \dots & \dots & 0 \end{bmatrix} \text{ (matriz de rotación circular)}$$

$$D_j = \begin{bmatrix} r_{ij} & & & & \\ & \ddots & & & \\ & & r_{j-1,j} & & 0 \\ & & & r_{j+1,j} & \\ 0 & & & & \ddots \\ & & & & & r_{N,j} \end{bmatrix}^{-1}$$

Por lo tanto la solución estimada sería,

$$\mathbf{x}_s = (\mathbf{S}_j^T * \mathbf{M}_j^T * \mathbf{M}_j * \mathbf{S}_j)^{-1} * \mathbf{S}_j^T * \mathbf{M}_j^T * \mathbf{M}_j * \boldsymbol{\mu}_j$$

3.1.2.6. Método de Levenberg - Marquardt

El método de Levenberg-Marquardt (LM) no es un método de resolución de ecuaciones TDOA directamente; es un método de optimización de ecuaciones no lineales. Este método es útil para resolver directamente las ecuaciones TDOA que se explicaron en el apartado de descripción matemática.

El método de LM resuelve los problemas de mínimos cuadrados no lineales al igual que el conocido método de Gauss-Newton. El problema de Gauss-Newton es que puede tener dificultades en algún punto del proceso iterativo si la matriz Jacobiana no tiene rango completo o está mal condicionada; debido a ello se creó el algoritmo de LM.

LM es por tanto un algoritmo más robusto que puede hallar una solución incluso cuando el punto inicial del proceso iterativo es lejano al punto mínimo final. Como ya se ha comentado, LM resuelve los problemas de mínimos cuadrados:

$$f(x) = \frac{1}{2} \sum_{j=1}^m r_j^2(x)$$

Donde $x = (x_1, x_2, \dots, x_n)$ es un vector que incluye las variables, cada r_j es una función y se denominan residuales y se asume que $m > n$. Para facilitar las cosas también se puede definir el siguiente vector $r(x)$ y así representar la función $f(x)$ como un vector residual :

$$r(x) = (r_1(x), r_2(x), \dots, r_m(x))$$

$$f(x) = \frac{1}{2} \|r(x)\|^2$$

Por otro lado, la matriz Jacobiana de r se puede escribir de la siguiente forma:

$$J(x) = \frac{\partial r_j}{\partial x_i} \text{ donde } 1 \leq j \leq m, 1 \leq i \leq n$$

Como ya se ha visto, las ecuaciones TDOA que se deben resolver son ecuaciones no lineales por lo que el estudio se va a centrar solamente en el caso de que cada función r_j sea no lineal.

Partiendo de lo anterior, las derivadas de la función a minimizar son:

$$\nabla f(x) = \sum_{j=1}^m r_j(x) * \nabla r_j(x) = J(x)^T * r(x)$$

$$\nabla^2 f(x) = J(x)^T * J(x) + \sum_{j=1}^m r_j(x) * \nabla^2 r_j(x)$$

La propiedad distintiva de los problemas de mínimos cuadrados es que dada la matriz Jacobiana J , se puede obtener la matriz Hessiana ($\nabla^2 f(x)$) de manera directa si se supone que los residuales ($r_j(x)$) son pequeños. De esta forma la matriz Hessiana quedaría de la siguiente forma:

$$\nabla^2 f(x) = J(x)^T * J(x)$$

Cabe destacar que los problemas en los que los residuales sean grandes, esta aproximación supondrá que este método no obtenga un buen rendimiento.

Como ya se ha comentado previamente el método LM apareció como una mejora del método de Newton, pero antes de nada se debe ver en orden cronológico cual fue la aportación de cada uno, Levenberg y Marquardt, al problema.

Levenberg propuso abordar el problema como una mezcla entre el método de Gradient Descent (que explicaremos más adelante) y el método de Newton.

Este método iterativo híbrido combina las ventajas de ambos métodos siguiendo la siguiente regla de actualización:

$$\mathbf{x}_{i+1} = \mathbf{x}_i - (\mathbf{H} + \lambda \mathbf{I})^{-1} \nabla f(\mathbf{x}_i)$$

Donde \mathbf{H} se corresponde con la matrix Hessiana evaluada en \mathbf{x}_i y λ es el tamaño del paso del método de Gradient Descent.

La regla de actualización funciona de la siguiente forma. Si el error disminuye significa que la suposición realizada funciona y por lo tanto se reduce el valor de λ , normalmente en un factor 10, para reducir la influencia del método Gradient Descent. Por el contrario, en caso de que el error crezca, se incrementa el valor de λ para obtener una mayor influencia del método de Gradient Descent.

Marquardt observó que en el caso de que λ sea elevado, la matriz Hessiana no se utiliza. Por ello, hizo un cambio en la regla de actualización anterior sustituyendo la matriz identidad por la diagonal de la matriz Hessiana para mejorar el acercamiento en cada iteración.

$$\mathbf{x}_{i+1} = \mathbf{x}_i - (\mathbf{H} + \lambda \text{diag}[\mathbf{H}])^{-1} \nabla f(\mathbf{x}_i)$$

Se observa fácilmente la aportación de cada método combinado, ya que ajustando λ a 0 se obtiene la regla de actualización del método de Newton.

$$\mathbf{x}_{i+1} = \mathbf{x}_i - \mathbf{H}^{-1} * \nabla f(\mathbf{x}_i)$$

3.1.2.7. Método Divide-and-Conquer

El método de Divide-and-Conquer (DAC) [10], presentado por J. S. Abel, consiste en dividir las medidas TDOA en diferentes grupos. La condición clave es que cada grupo debe tener un tamaño igual al número de incógnitas por lo que sólo se puede usar un número de medidas múltiplo del número de las incógnitas, desechando el resto. Se calcula una solución por cada grupo para luego combinarlas adecuadamente y obtener una solución final.

Este requisito clave que provoca una ineficiencia en cuanto a la utilización de medidas redundantes sitúa a este método en una clara desventaja frente a otros y por tanto no será fruto de estudio en este proyecto.

3.1.2.8. Método de Steepest Descent

El método de Steepest Descent, también conocido como Gradient descent, es básicamente un algoritmo de optimización de primer orden para hallar el mínimo local de una función. Ya se vió anteriormente que tiene una aportación muy importante en el método de Levenberg-Marquardt.

Este método basado en el gradiente no debe ser confundido con el método de aproximación utilizado para la resolución de integrales complejas. El Gradient Descent también puede ser usado para la resolución de un sistema de ecuaciones no lineales y de ahí su importancia en este proyecto. Con este propósito, este algoritmo fue aplicado para la resolución de ecuaciones TDOA por Lingwen Zhang y Zhenhui Tan, en 2008 [11].

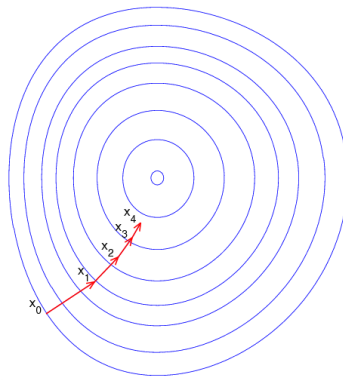


Figura 3.10: Método de Steepest Descent

La idea surgió gracias al problema de la estimación inicial del método de Taylor, explicado anteriormente. Ya se conoce la dependencia del método de Taylor con el punto inicial de iteración por lo que lo que se pretendía era realizar el método en dos fases. La primera fase consistiría en aplicar el método de Steepest Descent para obtener un punto estimado cercano al objetivo y en la segunda fase aplicar el método de Taylor con esa estimación inicial y así obtener una solución mucho más precisa. Este método se denominó Algoritmo de Optimización Híbrido (HOA).

No tiene ningún requisito en cuanto al número de ecuaciones por lo que es válido para cualquier número de antenas superior a 3, requisito necesario para determinar una posición en 3 dimensiones.

A continuación se expone la descripción matemática del método de Gradient Descent. Todo se inicia con la definición de la función diferencia de tiempos de llegada.

$$f_i(x, y, z) = (\sqrt{(x - x_i)^2 + (y - y_i)^2 + (z - z_i)^2} - \sqrt{(x - x_1)^2 + (y - y_1)^2 + (z - z_1)^2})$$

$i = 2, \dots, N$

O lo que es lo mismo

$$f_i(x, y, z) = \hat{d}_{i,1} + \varepsilon_{i,1}$$

$$i = 2, \dots, N$$

Siendo

$$\hat{d}_{i,1} = c(\hat{t}_i - \hat{t}_1) = \text{diferencia de distancias}$$

$$\varepsilon_{i,1} = \text{error de la estimación de diferencia de distancias}$$

La función puede ser reescrita de la siguiente forma:

$$\varphi(x, y, z) = f_i(x, y, z) - (\hat{d}_{i,1} + \varepsilon_{i,1})$$

Definiendo la siguiente curva a partir de la función anterior:

$$\Phi(x, y, z) = \sum_{i=2}^N [\varphi(x, y, z)]^2$$

El problema ahora se ha convertido en hallar el punto mínimo de Φ de manera iterativa. El procedimiento teórico es ir disminuyendo $\Phi(x, y, z)$, a partir de una estimación inicial, en la dirección de descenso más rápido (steepest descent) hasta obtener una solución. La dirección de steepest descent es la dirección opuesta al vector gradiente de $\Phi(x, y, z)$ que se puede denotar por,

$$G = \left(\frac{\partial \Phi}{\partial x}, \frac{\partial \Phi}{\partial y}, \frac{\partial \Phi}{\partial z} \right)^T$$

Conociendo la teoría, los pasos a seguir para hallar el mínimo de la función, formada a partir del sistema de ecuaciones TDOA, serían los siguientes.

1. Se parte de una estimación inicial (x_0, y_0, z_0)
2. Se calcula el vector gradiente en ese punto

$$G_0 = (g_{10} g_{20} g_{30})^T$$

Siendo

$$\begin{cases} g_{10} = \frac{\partial \Phi}{\partial x} \Big|_{(x_0, y_0, z_0)} = 2 \left[\sum_{i=2}^N \left(\frac{\partial \varphi_i}{\partial x} \right) \varphi_i \right]_{(x_0, y_0, z_0)} \\ g_{20} = \frac{\partial \Phi}{\partial y} \Big|_{(x_0, y_0, z_0)} = 2 \left[\sum_{i=2}^N \left(\frac{\partial \varphi_i}{\partial y} \right) \varphi_i \right]_{(x_0, y_0, z_0)} \\ g_{30} = \frac{\partial \Phi}{\partial z} \Big|_{(x_0, y_0, z_0)} = 2 \left[\sum_{i=2}^N \left(\frac{\partial \varphi_i}{\partial z} \right) \varphi_i \right]_{(x_0, y_0, z_0)} \end{cases}$$

3. Se elige un valor del parámetro λ que define el tamaño del paso en la dirección de $-G_0$ para hallar el siguiente punto (x_1, y_1, z_1) . El desarrollo teórico para hallar este valor de λ está en [9].

$$\lambda = \left[\frac{\sum_2^N \varphi_i \left(g_{10} \frac{\partial \varphi_i}{\partial x} + g_{20} \frac{\partial \varphi_i}{\partial y} + g_{30} \frac{\partial \varphi_i}{\partial z} \right)}{\sum_2^N \left(g_{10} \frac{\partial \varphi_i}{\partial x} + g_{20} \frac{\partial \varphi_i}{\partial y} + g_{30} \frac{\partial \varphi_i}{\partial z} \right)^2} \right]_{(x_0, y_0, z_0)}$$

4. Por último se calcula el siguiente punto de la siguiente forma

$$\begin{cases} x_1 = x_0 - \lambda g_{10} \\ y_1 = y_0 - \lambda g_{20} \\ z_1 = z_0 - \lambda g_{30} \end{cases}$$

5. Para comprobar cuando finalizar este proceso iterativo, es necesario calcular el valor de Φ , de modo que cuando tiende a 0 finaliza el proceso. En caso de estar lejos de ese valor objetivo, la siguiente iteración comienza con (x_1, y_1, z_1) y se repite el proceso.

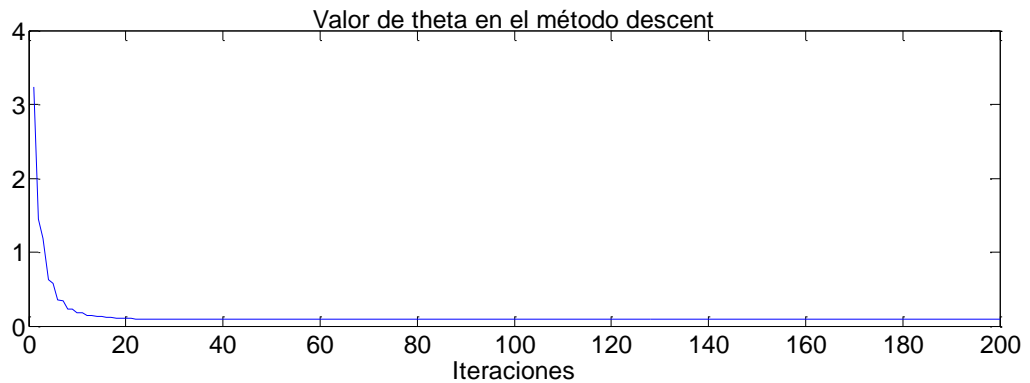


Figura 3.11: Variación del valor de theta en el método de Steepest Descent

3.1.3. Tabla comparativa

A continuación se muestra una tabla comparativa de los algoritmos comentados anteriormente.

Algoritmo	Solución	Medidas redundantes	Carga computacional	Comentarios
Método de Taylor	Óptima	Sí	Alta	<ul style="list-style-type: none"> - Requiere estimación inicial para el proceso iterativo - Problemas de convergencia
Método de Fang	Closed-form exact solution	No	Baja	<ul style="list-style-type: none"> - Sólo funciona para 4 antenas - Según método original, requisitos en cuanto a la posición de las antenas
Interpolación esférica	Subóptima	Si	Baja	<ul style="list-style-type: none"> - Requiere 5 antenas como mínimo
Método de Chan	Closed-form exact solution	Sí	Baja	<ul style="list-style-type: none"> - Algoritmo diferentes para más de 4 antenas
Método de Friedlander	Subóptima	Sí	Baja	<ul style="list-style-type: none"> - Permite ponderación de tiempos de llegada
Levenberg-Marquardt	Heurística	Sí	Baja	<ul style="list-style-type: none"> - Requiere estimación inicial para el proceso iterativo
Steepest Descent	Mínimo local	Sí	Baja	<ul style="list-style-type: none"> - Requiere estimación inicial para el proceso iterativo

Tabla 3.1: Resumen de algoritmos de estimación

3.2. RECEIVED SIGNAL STRENGTH

La técnica RSS se basa en las potencias recibidas de las señales para efectuar el posicionamiento. En caso de que el target sea el transmisor de las señales, los sensores obtendrán las potencias recibidas y se encargarán de enviar estos datos a un punto central. En caso contrario, el target será el que obtenga las potencias de las señales emitidas por los sensores y el que debe realizar los cálculos de auto-posicionamiento. Destacar que la segunda opción es la más versátil por sencillez y ahorro económico.

Los valores de potencia recibidos junto al conocimiento previo de la posición de los sensores son suficientes para estimar la posición del target aunque también existen otras formas como utilizar mapas de potencias. A continuación se verán diferentes técnicas utilizadas a partir de los valores de RSS.

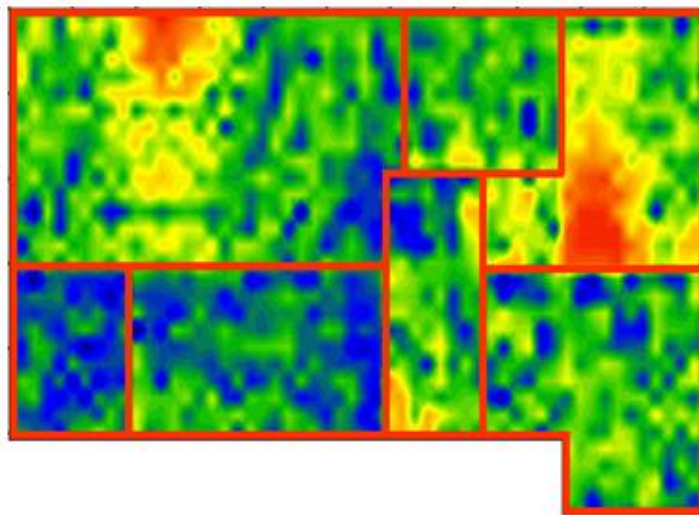


Figura 3.12: Ejemplo de mapa de potencias en una vivienda

3.2.1. Métodos de estimación de localización basados en RSS

Principalmente se pueden diferenciar dos tipos de métodos; localización por triangulación y localización por reconocimiento de patrones. El primero de ellos utiliza las mismas ideas explicadas anteriormente en el capítulo de TDOA.

3.2.1.1. Triangulación

El método de triangulación parte de las intensidades de las señales y las convierte en distancias o tiempos, sabiendo obviamente la potencia de los transmisores. El problema de este método es saber qué modelo de interiores se va a utilizar para estimar las distancias y no obtener unos valores poco precisos.

La localización se basa en la suposición de que las antenas generan un patrón de radiación en forma de esfera con centro en la propia antena. Con cuatro de estas esferas,

cada una de ellas con un radio estimado en base a la potencia recibida de cada una de estas cuatro antenas, se obtiene la estimación de la localización al intersecar las cuatro esferas.

Esto parte de la suposición de que se haga uso de una antena con patrón de radiación esférico; sin embargo, las antenas reales no pueden generar un patrón de radiación esférico, y por ello se usan antenas omnidireccionales, que tienen un patrón de radiación en forma de donut.

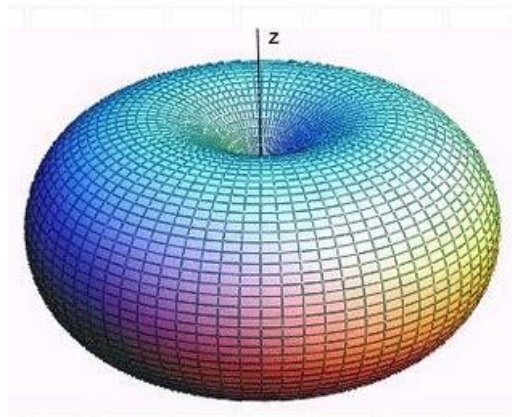


Figura 3.13: Patrón de radiación de antena omnidireccional

La ventaja de este método es que es de aplicación directa en cualquier entorno, suponiendo que el modelo empleado en cada caso se ajuste a la propagación real. Si la propagación real difiere mucho de la teórica, se tendrá que obtener un modelo adaptado del entorno por lo que la generalidad, ventaja principal del método, se pierde.

3.2.1.2. Reconocimiento de patrones

Las técnicas de reconocimiento de patrones, o fingerprinting, intentan encajar el vector de potencias recibidas (RSS) procedente de varios sensores, con una muestra de calibración sin necesidad de usar algoritmos geométricos. Es un método muy sencillo pero requiere de un trabajo previo importante.

Para su uso, primero hay que obtener un radio mapa patrón, que en este caso es una base de datos con posiciones en el edificio y los correspondientes niveles RSS recibidos de cada sensor en esas posiciones. La RSS se puede calcular de forma teórica, con el uso de un modelo de predicción de potencias, o bien se puede obtener de forma práctica viendo en cada punto de interés la potencia recibida.

Un parámetro importante es la resolución de las medidas o grado de parametrización; es decir el número de posiciones en el edificio que vamos a discretizar. Si por ejemplo se quiere tener medidas cada 50cm en un edificio de tamaño medio, ocurrirán dos cosas. La primera es que el trabajo de construcción del mapa radio será muy tedioso y largo; y por otro lado, la base de datos puede alcanzar un tamaño

considerablemente grande. El compromiso de precisión frente a tiempo es muy importante. Además, otro posible problema es que un cambio en el escenario (una obra, un cambio de posición de un sensor...) provoca la invalidez del mapa radio y por tanto la repetición de todo el proceso previo de medida.

La ventaja de este método frente a la triangulación es que los cálculos son más reducidos, y a diferencia del anterior, la discretización del entorno hace que no sea posible obtener una localización que no se haya definido; en cambio, en la triangulación se puede localizar en cualquier punto del espacio por no haberse discretizado la planta.

Los cálculos que permiten la estimación de la localización en los algoritmos de reconocimiento de patrones son sencillos y cambian según sea el método a emplear, pero se basan en, dado un vector de potencias, encontrar la localización que tenga el vector o vectores de potencias más parecidos.

El mayor reto de estos algoritmos es diferenciar entre dos posiciones con vectores de potencias similares. Este problema es muy importante porque se pueden incurrir en grandes errores. Existen varias soluciones para este problema en caso de tener un target móvil que recorra un escenario porque se pueden hacer estimaciones lógicas de diferencia de distancias entre puntos consecutivos para rechazar posiciones muy lejanas. Sin embargo, en caso de un posicionamiento puntual no existen esas mejoras.

• Los métodos KNN

En los últimos años se han desarrollado varios tipos de algoritmos de RSS basados en el reconocimiento de patrones. Uno de ellos es el algoritmo KNN (K-Nearest Neighbors). KNN se basa en el recubrimiento de un escenario con sensores, con el requisito de que sus huellas de potencia se solapen unas con otras. La idea sobre la que se basa el algoritmo es que, aunque la RSS recibida en el target no depende linealmente de la distancia al sensor, sí que guarda alguna relación; al menos se puede asegurar que si el sensor está muy cerca del target la potencia recibida será alta, y si el sensor está lejos, la RSS disminuirá.

En base a esta simple idea, se crea el radio mapa explicado anteriormente que incluye las posiciones y los RSS de cada sensor. En la figura siguiente se muestra un ejemplo de cómo será esta base de datos.

Punto	Coordenada x	Coordenada y	Coordenada z	RSS ₁	RSS ₂	RSS ₃	RSS ₄
1	0.5	1	1.5	54.2	26.3	36.1	40.1
...

Tabla 3.2: Ejemplo de radio mapa patrón

Una vez el target se quiere localizar, se obtiene un vector de potencias x el cual se compara con la base de datos de localizaciones y se estima la localización viendo cuáles de todos los vectores de prueba son más parecidos al vector x .

El algoritmo KNN parte del mapa de potencias P y del vector de potencias x . El método consiste en calcular las distancias euclídeas entre el vector de potencias y cada elemento de la base de datos para ordenar dicha base de datos y así poder obtener los k vecinos más cercanos al vector de potencias de interés.

Particularizando a nuestra situación donde tenemos cada medida con tres coordenadas las matrices de inicio son las siguientes:

$$P = \begin{bmatrix} x_1 & y_1 & z_1 & RSS_{11} & RSS_{12} & RSS_{13} & \dots & RSS_{1n} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_m & y_m & z_m & RSS_{m1} & RSS_{m2} & RSS_{m3} & \dots & RSS_{mn} \end{bmatrix}$$

$$x = [RSS_{x1} \quad \dots \quad RSS_{xn}]$$

Donde m es el número de medidas y n es el número sensores. A partir de estos datos el vecino más cercano sería el que cumpla la mínima diferencia entre los RSS obtenidos y los del mapa de potencias.

$$d_j = \sqrt{\sum_{i=1}^n (x(i) - P(j, 3 + i))^2} \text{ siendo } j = 1, \dots, m$$

$$\text{Posición}_{\text{estimada}} = [P(j, 1) \quad P(j, 2) \quad P(j, 3)] \text{ siendo } j = \min(d_j)$$

Esta fórmula considera $k = 1$, es decir, un solo vecino es el que minimiza la distancia. En este caso concreto el algoritmo K-NN pasa a denominarse Nearest Neighbor Algorithm, es decir el algoritmo del vecino más cercano.

Una mejora de este método es la introducción de pesos una vez obtenidos los k vecinos, y combinarlos para obtener una localización intermedia entre ellos; a este enfoque se le denomina WKNN (Weighted K-Nearest Neighbors).

Estos dos algoritmos consiguen bajos errores de estimación, pero su inconveniente principal es el tamaño de la tabla de información de calibración y también el elevado tiempo de ejecución si hay una cantidad considerable de sensores y/o localizaciones.

• Métodos bayesianos

Una alternativa a la anterior, pero con características similares por su etapa previa, son métodos basados en redes bayesianas. Un ejemplo de ello es un proyecto

denominado Nibble de la universidad de Los Ángeles (UCLA) que se basa en redes WiFi.

Se basa en la construcción de una red bayesiana en la fase de entrenamiento, de forma que el usuario determina los puntos de interés, y posteriormente entrena durante un tiempo al sistema, que periódicamente recoge muestras de RSS. Por esta razón, el sistema Nibble es más inmune a los cambios intrínsecos que hay en Wi-Fi por los multitrayectos.

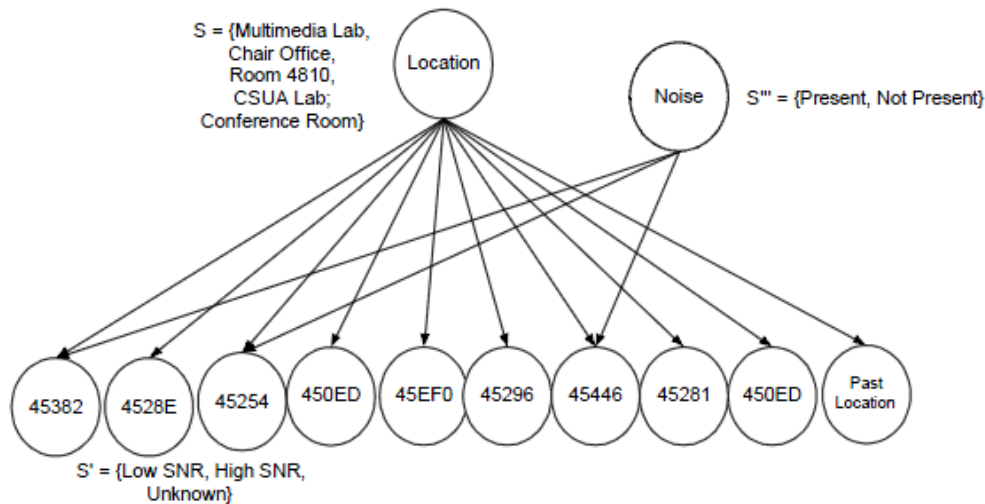


Figura 3.14: Ejemplo de red bayesiana del sistema Nibble

El sistema Nibble es esencialmente experimental, lo que quiere decir que no ha tenido en cuenta la modelización de la señal Wi-Fi, sino que tiene una fase de entrenamiento en la cual primero se definen la probabilidad a priori de estar en una localización $p(L)$, y después las probabilidades a posteriori, probabilidades condicionadas de recibir una determinada RSS de un punto de acceso estando en una localización determinada, $p(E|L)$.

En la figura 3.14 se muestra una red bayesiana que el sistema Nibble puede usar para inferir la posición del target. El nodo raíz de la red describe las probabilidades $p(L)$, mientras que los nodos terminales representan las probabilidades $p(E|L)$. Al obtener medidas del nivel sensor se hallan las probabilidades a posteriori, con las que se hallará la posición más probable del target dados esos datos.

La calibración de Nibble lleva mucho tiempo al igual que en el caso de KNN, porque hay que tomar varias muestras en días distintos en cada localización. En cuanto a las probabilidades a priori, por omisión siguen una distribución uniforme, pero se pueden incorporar perfiles de usuario si se desea modificar estas probabilidades.

Por último, el sistema Nibble también implementa un sistema de mejora en el caso de tener un target móvil. En este caso el sistema Nibble modela transiciones imposibles

de una localización a otra, mediante un modelo similar a un Modelo Oculto de Markov (HMM).

- **Métodos basados en redes neuronales**

La funcionalidad de las redes neuronales es clasificar datos en conjuntos, por ello es obvio que se pueden usar en la estimación de la localización mediante la observación de la RSS de los distintos puntos de acceso.

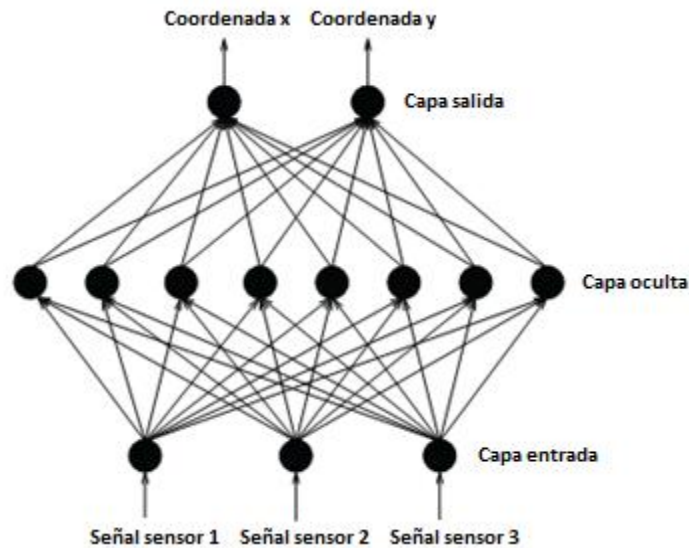


Figura 3.15: Perceptrón multicapa con una capa intermedia para localización 2D

El método empleado es un perceptrón multicapa, en el cual las entradas pueden ser la RSS de cada uno de los sensores, y las salidas son las probabilidades de estar en cada una de las localizaciones. Este método tarda mucho tiempo en entrenarse y necesita cantidades ingentes de datos de prueba, pero además corre el riesgo de sobreajustarse (overfitting), planificándose cuidadosamente cuál ha de ser el número de pesos a ajustar y cuántos casos de prueba se van a usar.

Otro problema es que, aunque la fase de entrenamiento se haya realizado correctamente (en número de iteraciones y de casos de prueba) no se garantiza que ante los datos reales la respuesta sea adecuada. Sin embargo, parece uno de los métodos más prometedores para estimar la localización del móvil en el posicionamiento en interiores.

3.3. RESULTADOS DE LAS SIMULACIONES

Una vez explicada la teoría el siguiente paso consiste en explicar el proceso que se ha llevado a cabo para comparar las diferentes técnicas de localización en interiores. Este proceso se ha desarrollado utilizando el entorno de programación Matlab.

Para comenzar, las simulaciones se pueden separar en dos partes claramente diferenciadas por la utilización de diferentes técnicas. Una primera parte con un mayor peso en el proyecto en la que se comparan diferentes algoritmos de estimación basados en TDOA; y una segunda parte en la que se implementa un algoritmo de estimación basado en el indicador RSS. Concretamente, el proyecto se ha centrado en los algoritmos más utilizados e implantados descartando algunos con ciertos requisitos que los sitúan en clara desventaja o por falta de precisión. A continuación se muestran los algoritmos implementados en este estudio.

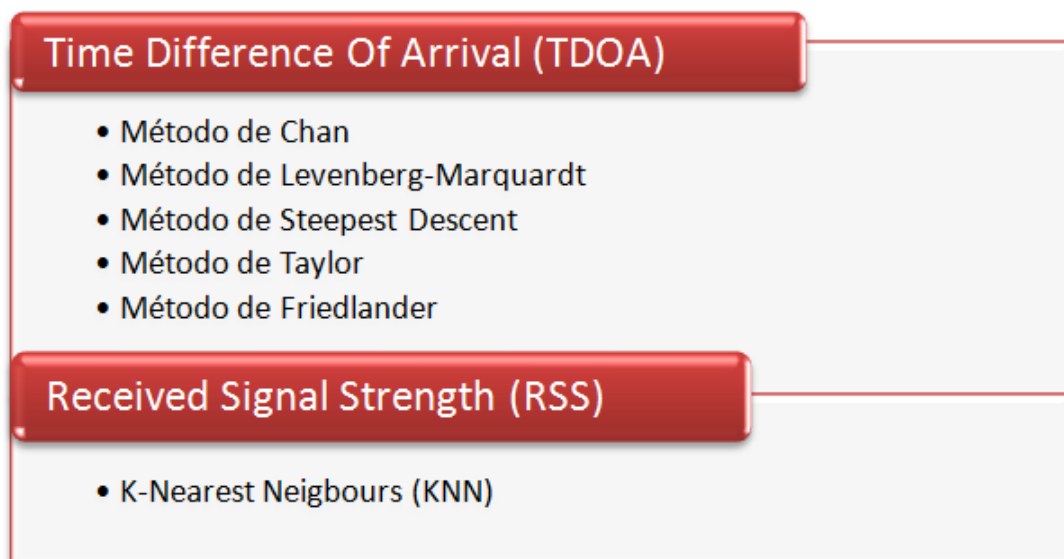


Figura 3.16: Métodos de localización implementados

Una vez implementados los diferentes tipos de algoritmos el siguiente paso consiste en comprobar su funcionamiento.

El proceso práctico que se ha llevado a cabo ha tenido dos enfoques distintos; el primero de ellos ha sido un estudio previo de los diferentes algoritmos en el que se ha comprobado su funcionamiento y el segundo enfoque ha consistido en un estudio más avanzado que ha partido de datos obtenidos con un algoritmo de trazado de rayos en tres dimensiones para entornos indoor [12].

3.3.1. Estudio preliminar

El estudio preliminar se basa en analizar las características principales de los algoritmos. Para dicho fin, se han supuesto escenarios rectangulares sin ningún tipo de obstáculo. Por lo tanto, las medidas de distancias y retardos entre las antenas y el target son exactas. En las simulaciones que se ha pretendido emular un escenario un poco más realista se ha añadido un término de ruido blanco Gaussiano con una SNR predeterminada a las medidas de distancias o retardos.

La primera evaluación pretende definir los rangos de precisión que se obtienen con estos algoritmos. Para dicho cometido se realizaron dos simulaciones en las que se utilizaron dos escenarios ideales; un escenario grande con dimensiones 50x20x4 metros que se correspondería con una nave industrial y otro más pequeño simulando una casa con dimensiones 12x6x4 metros. En cada escenario se han realizado 100 iteraciones con las antenas y el target en posiciones aleatorias y se han calculado los errores de los algoritmos.

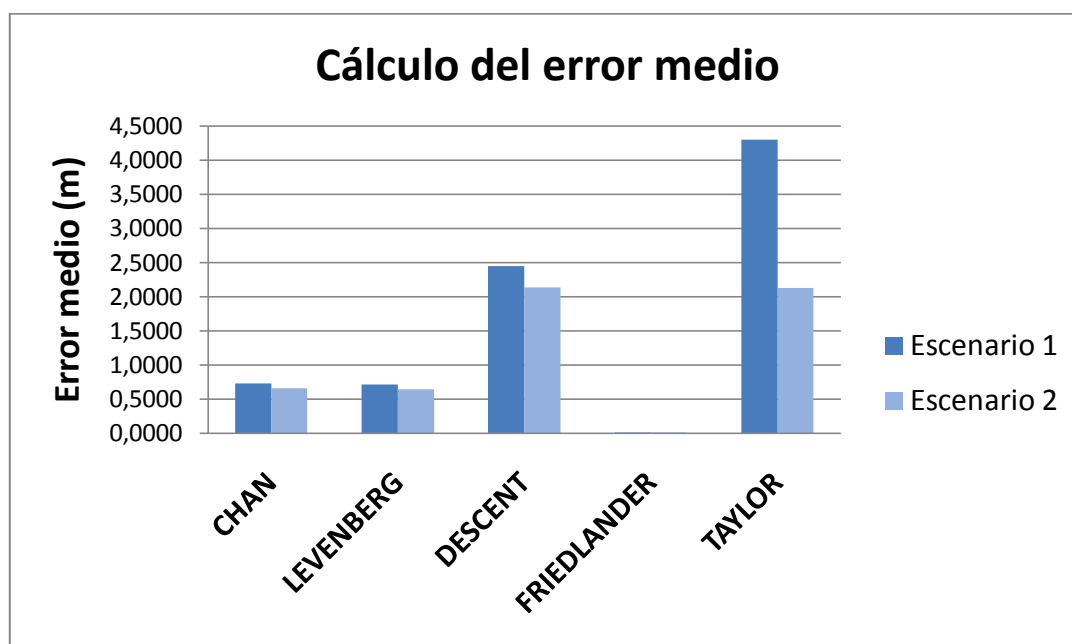


Figura 3.17: Error medio de los diferentes algoritmos

Se observa que el error medio oscila entre 0.6 y 4.25 metros para el escenario grande (50x20x4) y entre 0.5 y 2.1 metros en el escenario pequeño (12x6x4); salvo en el caso del método de Friedlander que su error es aproximadamente 0.

Por otro lado es necesario hacer una distinción entre los algoritmos que requieren una estimación previa y los que no. Se comprueba que los algoritmos que la requieren como son Levenberg, Descent y Taylor, son los métodos con peor precisión y además el error es mayor cuanto mayor es el escenario.

En la siguiente figura se muestran los resultados anteriores del escenario grande (50x20x4) con un poco más de detalle. Se han creado tres rangos de error, tal como se ve en la leyenda de la imagen, y se ha calculado el porcentaje de cada uno para cada algoritmo.

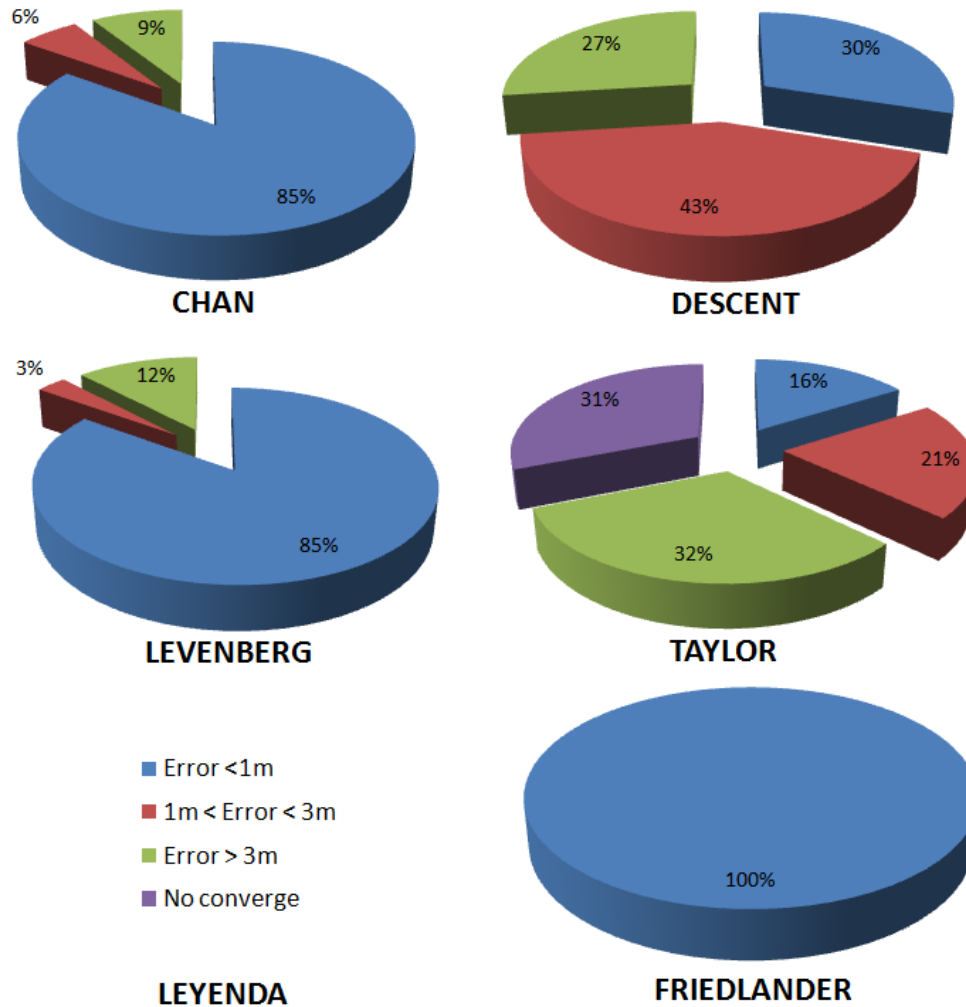


Figura 3.18: Rangos de error en el escenario grande (50x20x4)

La siguiente simulación pretende mostrar la dependencia del error de las estimaciones respecto al error introducido en las medidas. En un escenario con muchos obstáculos la señal sufre múltiples rebotes que alteran considerablemente la exactitud de las medidas obtenidas por el target. En este caso, se ha simulado un escenario con unas dimensiones de 100x100x4 metros. Para introducir un error en las medidas se ha añadido a las mismas un término de ruido que se ha ido aumentando.

La siguiente figura muestra el incremento del error de los algoritmos más fiables vistos hasta ahora, cuando se disminuye la SNR del término de ruido añadido. Para cada valor de ruido introducido se ha realizado una simulación con 100 iteraciones y se ha calculado el error medio de las estimaciones de los algoritmos.

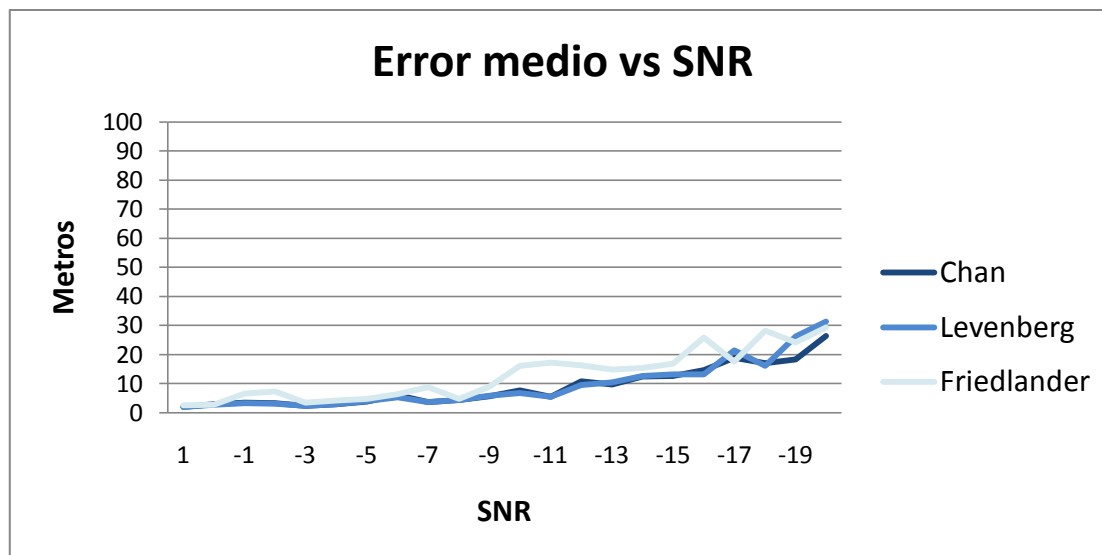


Figura 3.19: Error medio vs SNR

Las conclusiones más relevantes que se pueden extraer de esta figura son que los algoritmos de Levenberg y Friedlander se comportan de manera similar ante el empeoramiento de las medidas mientras que el método de Friedlander sufre más con medidas inexactas; es decir en un entorno más realista.

Tal como se ha visto con anterioridad, el punto inicial en varios métodos de localización es su gran punto débil. Para comprobar dicha afirmación se ha realizado la siguiente gráfica. Para ello, el proceso que se ha realizado ha sido simular un escenario ideal es decir con medidas exactas para evitar la influencia de otros factores. Las dimensiones elegidas para dicho escenario han sido 20x20x4 metros y se han ido tomando puntos iniciales cada vez más cercanos al target a localizar. Se ha procedido a calcular el error medio de las estimaciones y finalmente los resultados obtenidos se muestran en la figura 3.20.

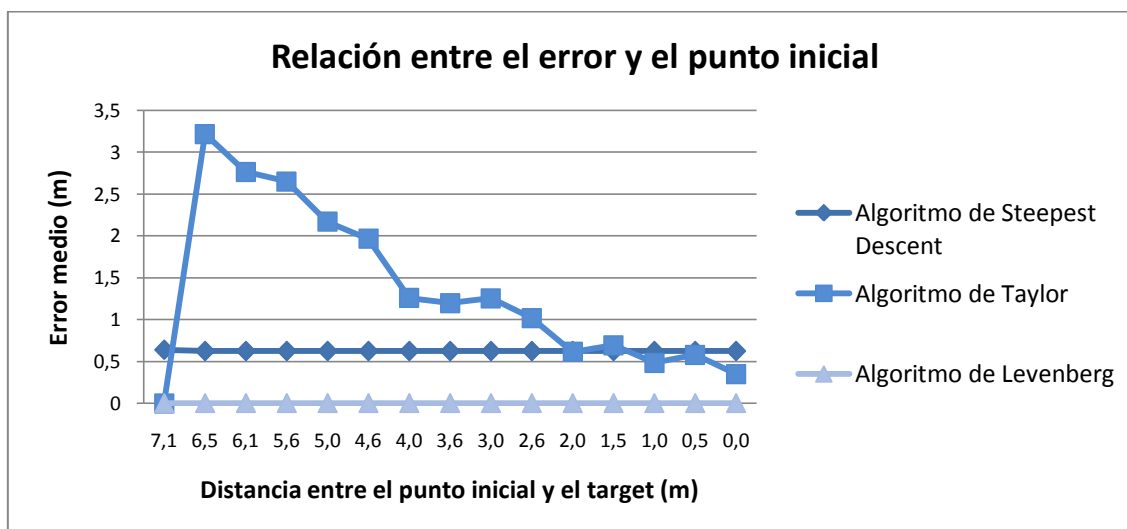


Figura 3.20: Dependencia de las estimaciones con el punto inicial de iteración

Las distancias elegidas no son triviales ya que en dicho escenario a partir de 7 metros el algoritmo de Taylor ya no converge (primer punto de la gráfica). A partir de ese valor, se observa como el error de la estimación por Taylor disminuye al aproximarse el punto inicial a la posición real del target. Los otros dos algoritmos mantienen un error constante en el rango estudiado por lo que son mucho más eficaces ante este efecto.

En definitiva, el método de Taylor es muy sensible al requisito del punto inicial y por tanto son necesarias soluciones alternativas. Una solución bastante eficaz es utilizar como punto inicial la estimación de otro algoritmo. Dado que el método de Descent o Levenberg son dependientes del punto inicial, se ha procedido a incluir en el estudio posterior un algoritmo alternativo basado en el método de Chan que a partir de ahora denominaremos Taylor-Chan.

Este nuevo método, nacido como mejora del algoritmo de Taylor, tiene dos fases; cálculo de la estimación chan y cálculo de la estimación Taylor con la estimación anterior como punto de partida. Con este método se consigue un mayor porcentaje de convergencia ya que se reduce el efecto adverso del punto inicial aunque ahora se depende completamente de la precisión del algoritmo de Chan.

3.3.2. Estudio avanzado

El estudio avanzado hace uso de un algoritmo, desarrollado en el entorno de simulación Matlab, fruto de un proyecto final de carrera en la Universidad Pública de Navarra. Mediante este algoritmo de trazado de rayos en tres dimensiones se puede caracterizar la radiopropagación electromagnética en entornos indoor.

El algoritmo tiene tres fases durante su ejecución:

- Fase 1: Creación del escenario
- Fase 2: Simulación de trazado de rayos en tres dimensiones
- Fase 3: Análisis de los resultados

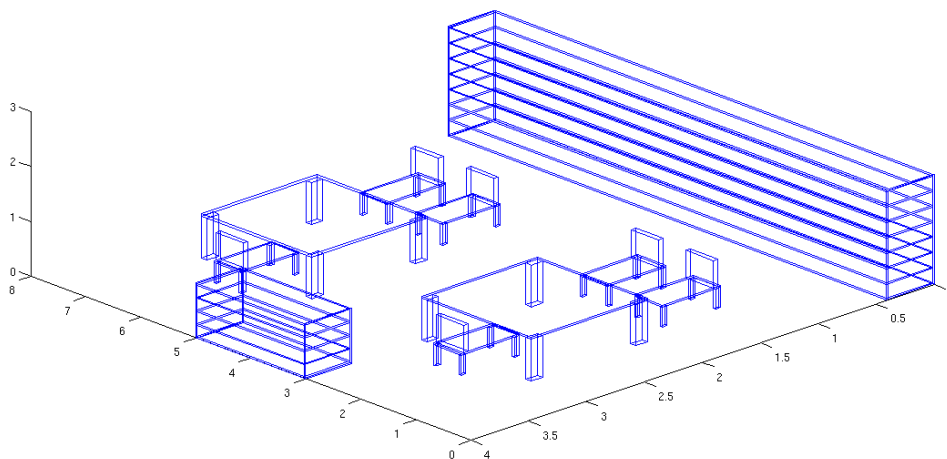


Figura 3.21: Escenario con varios objetos

Para el desarrollo de la fase 2 es necesario asignar valores a ciertos parámetros relacionados con el trazado de rayos como pueden ser las variaciones en ángulos del barrido, el número de rebotes de las señales, la frecuencia o la resolución de las medidas obtenidas que afectarán a los resultados del algoritmo.

Conociendo el objetivo y requisitos de este proyecto, estos valores se han fijado de la siguiente manera:

Parámetro	Nomenclatura	Valor
Ángulo Phi inicial	$\varphi_{inicial}$	0
Incremento ángulo Phi	$\Delta\varphi$	$2 * (\pi/180)$
Ángulo Phi final	φ_{final}	$2 * \pi - 2 * (\pi/180)$
Ángulo Theta inicial	$\theta_{inicial}$	0
Incremento ángulo Theta	$\Delta\theta$	$\pi/90$
Ángulo Theta final	θ_{final}	$\pi - \pi/90$
Frecuencia (GHz)	f	2.4
Número de rebotes	n	6
Resolución eje x (m)	Δx	0.2
Resolución eje y (m)	Δy	0.2
Resolución eje z (m)	Δz	0.2
Potencia transmitida (mW)	P_{tx}	1

Tabla 3.3: Parámetros del algoritmo del trazado de rayos

Los datos de interés para este proyecto son los valores de potencia y retardo (al fin y al cabo son los indicadores RSS y TOA respectivamente) que se obtienen al ejecutar el algoritmo. De esta forma, por cada antena que se desee introducir en el escenario, se le asigna una posición en el mismo y se ejecuta el algoritmo.

Una vez obtenidos los datos para las diferentes antenas en un escenario ya sólo quedaría situar el target en la posición que se desee, extraer los datos de potencia o retardo recibidos por cada antena según la técnica que se desee utilizar y por último ejecutar los algoritmos de localización. En la figura 3.22 se muestra un mapa de potencias de un escenario con dos antenas WiFi; resultante de los datos obtenidos con el algoritmo utilizado.

Una vez conocido el proceso de extracción de datos el siguiente paso consiste en estudiar los diferentes entornos analizados con el algoritmo y por tanto sujetos posteriormente al análisis de localización concreto de este proyecto. En total se han utilizado seis escenarios que representan una casuística bastante completa y por lo tanto permiten realizar una adecuada comparación de resultados.

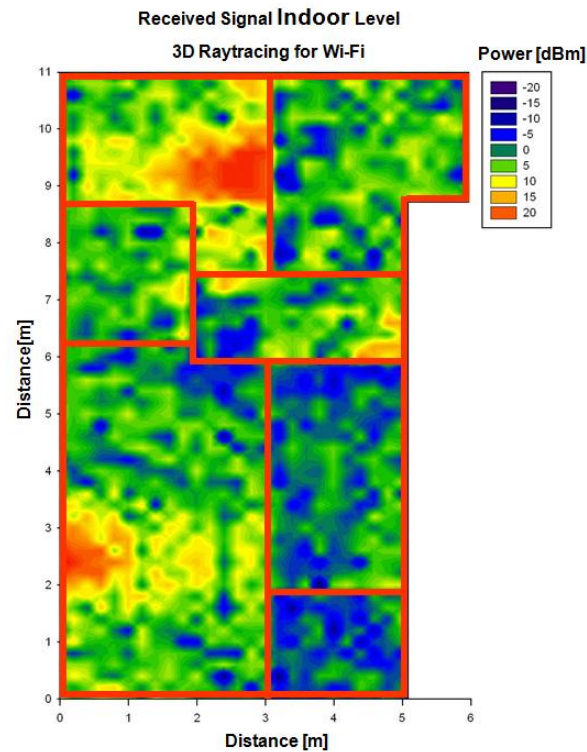


Figura 3.22: Mapa de potencia utilizando dos antenas WiFi

3.3.2.1. Descripción de los escenarios

■ Escenario Carlos III

Este escenario es un piso localizado en Madrid realizado con la colaboración del instituto de salud Carlos III. Consta de cinco habitaciones y por lo tanto representa un escenario muy complejo y por lo tanto muy representativo.

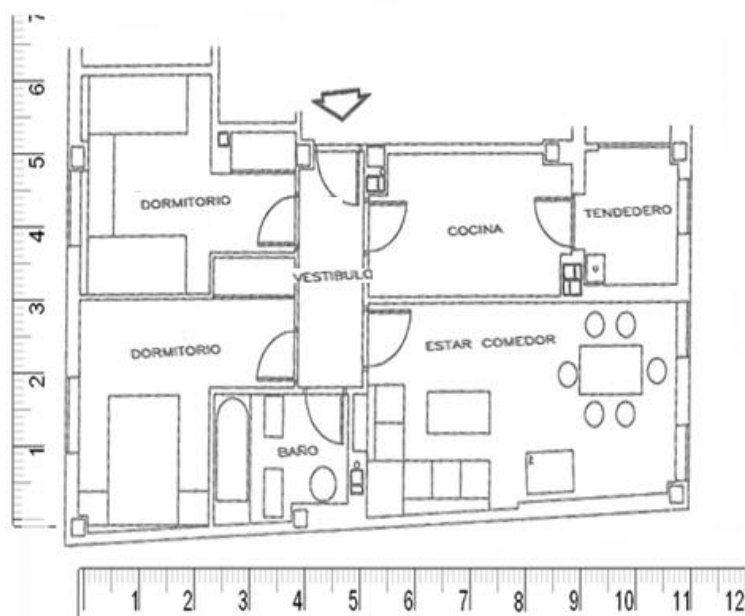


Figura 3.23: Escenario Carlos III

- Escenario Jerónimo de Ayanz I

Este escenario es un laboratorio vacío del nuevo centro de I+D en Electrónica y Comunicaciones Jerónimo de Ayanz.

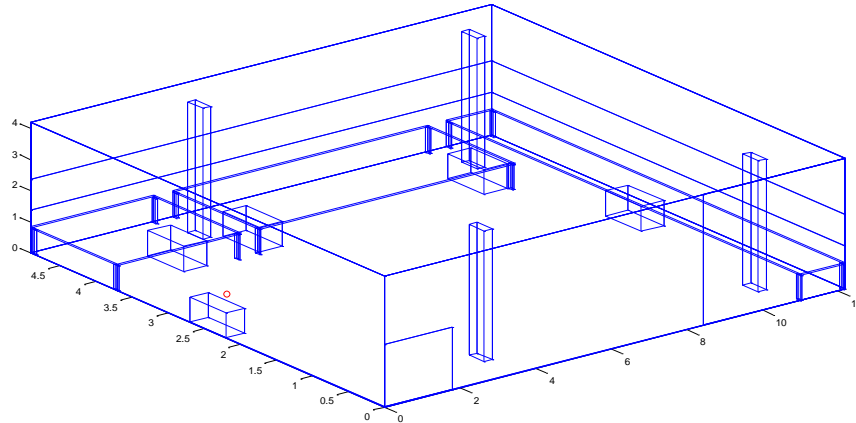


Figura 3.24: Escenario Jerónimo de Ayanz I

- Escenario Jerónimo de Ayanz II

Este escenario es otro laboratorio del nuevo centro de I+D en Electrónica y Comunicaciones de la UPNA Jerónimo de Ayanz.

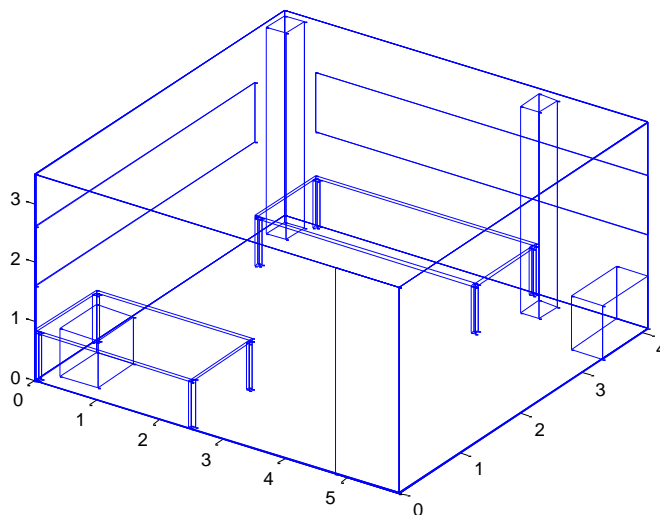


Figura 3.25: Escenario Jerónimo de Ayanz II

- Escenario SB

Este escenario se corresponde con la sala de becarios del departamento de Ingeniería Eléctrica y Electrónica ubicada en el edificio de los Tejos.

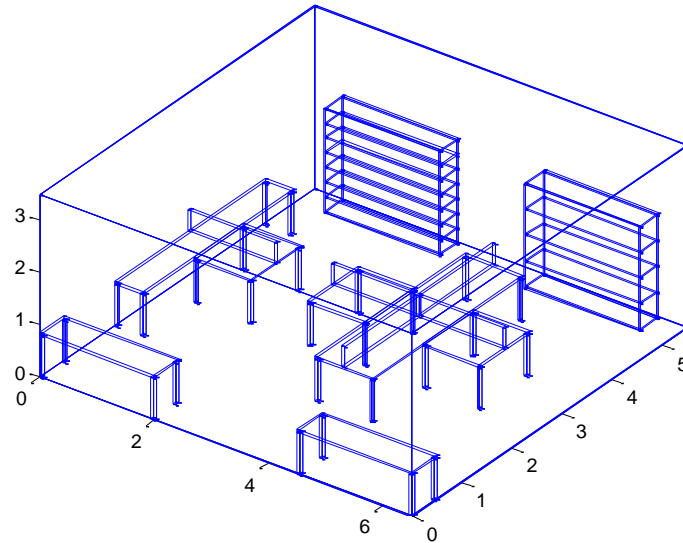


Figura 3.26: Escenario SB

- Escenario HVC

Este escenario se corresponde con una sala de maternidad del Hospital Virgen del Camino.

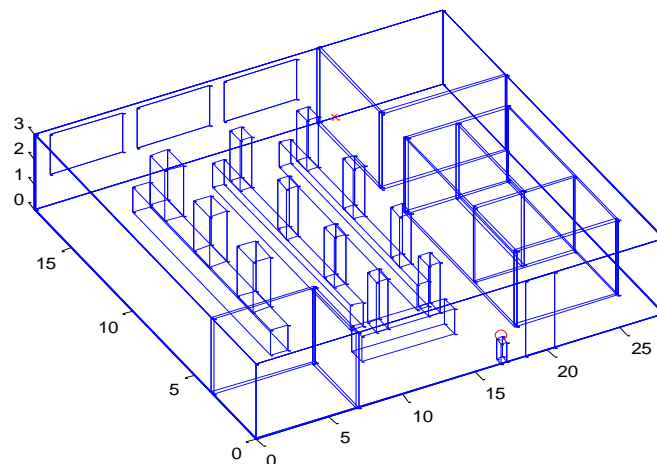


Figura 3.27: Escenario HVC

- Escenario LTE

Este pequeño escenario está inspirado en la forma de cualquier despacho de la UPNA simulando una femtocelda LTE.

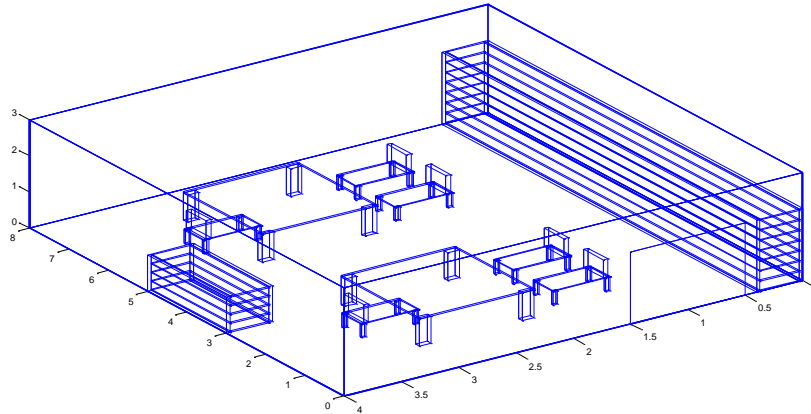


Figura 3.28: Escenario LTE

3.3.2.2. Resultados obtenidos

En este apartado se van a diferenciar las dos técnicas estudiadas en este proyecto. En primer lugar se exponen los resultados obtenidos para la técnica TDOA que es la parte más amplia y por último se muestran los resultados obtenidos con RSS.

3.3.2.2.1. TDOA

Los primeros resultados obtenidos han consistido en situar en cada escenario diferentes targets y calcular las posiciones estimadas por los diferentes algoritmos estudiados.

Los algoritmos de Levenberg, Descent y Taylor requieren una estimación inicial tal como se ha mostrado en la descripción matemática y en el estudio preliminar. Con los resultados obtenidos en dicho estudio, se ha decidido utilizar el centro de cada escenario como valor inicial para dichos algoritmos. Las medidas del escenario deben ser conocidas de antemano por lo que es el punto más equitativo para comenzar a iterar.

Tal como se mencionó en el apartado anterior, además de utilizar el algoritmo de Taylor se va a incluir el algoritmo Taylor-Chan para comparar ambos algoritmos.

Se han realizado simulaciones para 4 y 5 antenas en cada escenario y se han calculado tres tipos de parámetros muy similares (error medio, SEP y DRMS) explicados en el apartado 3.1.1, para calcular la precisión de los algoritmos. Además se ha añadido el porcentaje de convergencia de los algoritmos principalmente para poder comparar el método de Taylor en igualdad con el resto de métodos.

A continuación se exponen los resultados obtenidos:

Algoritmo	Error total	Error eje x	Error eje y	Error eje z	Porcentaje de convergencia	SEP	DRMS
Chan	1,62	0,96	1,21	0,29	100,00%	1,25	1,57
Levenberg	1,59	0,80	1,16	0,48	100,00%	1,25	1,49
Descent	1,36	0,73	1,04	0,29	100,00%	1,05	1,30
Taylor	1,41	0,42	0,85	0,51	30,77%	0,91	1,08
Taylor chan	1,44	0,63	0,75	0,48	46,15%	0,95	1,09
Friedlander	2,62	1,01	1,68	0,61	100,00%	1,68	2,05

Tabla 3.4: Carlos III 4 antenas

Algoritmo	Error total	Error eje x	Error eje y	Error eje z	Porcentaje de convergencia	SEP	DRMS
Chan	3,11	0,95	1,25	2,27	100,00%	2,28	2,76
Levenberg	1,69	0,70	0,97	0,85	100,00%	1,29	1,47
Descent	1,52	0,69	0,93	0,62	100,00%	1,14	1,31
Taylor	1,29	0,31	0,87	0,66	69,23%	0,94	1,14
Taylor chan	1,33	0,52	0,71	0,75	84,62%	1,01	1,15
Friedlander	3,82	0,87	0,88	3,41	100,00%	2,63	3,63

Tabla 3.5: Carlos III 5 antenas

Algoritmo	Media error (m)	Error eje x	Error eje y	Error eje z	Porcentaje de convergencia	SEP	DRMS
Chan	1,62	1,07	0,55	1,18	100,00%	1,43	1,69
Levenberg	1,71	1,05	0,54	1,05	100,00%	1,35	1,58
Descent	1,42	0,74	0,51	1,26	100,00%	1,28	1,55
Taylor	1,43	0,76	0,54	0,92	84,62%	1,13	1,30
Taylor chan	1,44	0,82	0,43	1,34	100,00%	1,32	1,63
Friedlander	1,37	0,61	0,36	1,03	100,00%	1,02	1,25

Tabla 3.6: Jerónimo de Ayanz I 4 antenas

Algoritmo	Error total	Error eje x	Error eje y	Error eje z	Porcentaje de convergencia	SEP	DRMS
Chan	2,58	1,59	0,58	1,30	100,00%	1,77	2,13
Levenberg	2,39	1,57	0,67	1,24	100,00%	1,77	2,11
Descent	1,05	0,60	0,47	0,61	100,00%	0,85	0,97
Taylor	1,53	0,72	0,43	1,14	75,00%	1,16	1,41
Taylor chan	1,33	0,69	0,44	0,82	91,67%	0,99	1,15
Friedlander	1,32	0,62	0,37	1,00	100,00%	1,01	1,23

Tabla 3.7: Jerónimo de Ayanz I 5 antenas

Algoritmo	Media error	Error Eje x	Error Eje y	Error Eje z	Porcentaje de convergencia	SEP	DRMS
Chan	1,71	0,55	0,47	1,45	100%	1,26	1,62
Levenberg	1,76	0,59	0,52	1,49	100%	1,32	1,68
Descent	1,26	0,42	0,30	1,10	100%	0,93	1,22
Taylor	0,93	0,19	0,26	0,81	90%	0,65	0,88
Taylor chan	0,93	0,18	0,36	0,78	100%	0,67	0,88
Friedlander	2,18	0,37	0,46	2,07	100%	1,48	2,15

Tabla 3.8: Jerónimo de Ayanz II 4 antenas

Algoritmo	Media error	Error Eje x	Error Eje y	Error Eje z	Porcentaje de convergencia	SEP	DRMS
Chan	1,25	1,07	0,26	0,37	100,00%	0,87	1,16
Levenberg	0,49	0,11	0,14	0,44	100,00%	0,35	0,48
Descent	0,60	0,15	0,20	0,51	100,00%	0,44	0,56
Taylor	0,47	0,23	0,16	0,30	100,00%	0,35	0,41
Taylor chan	0,68	0,28	0,12	0,55	80,00%	0,48	0,63
Friedlander	1,00	0,09	0,17	0,97	100,00%	0,63	0,99

Tabla 3.9: Jerónimo de Ayanz II 5 antenas

Algoritmo	Media error	Error Eje x	Error Eje y	Error Eje z	Porcentaje de convergencia	SEP	DRMS
Chan	2,01	1,56	0,78	0,62	100,00%	1,51	1,85
Levenberg	2,34	1,47	1,14	0,68	100,00%	1,68	1,98
Descent	2,06	1,16	1,03	0,96	100,00%	1,60	1,82
Taylor	2,44	1,54	1,45	0,90	54,55%	1,98	2,29
Taylor chan	2,02	1,33	1,05	0,74	63,64%	1,59	1,85
Friedlander	2,46	1,24	1,57	0,94	100,00%	1,91	2,21

Tabla 3.10: HVC 4 antenas

Algoritmo	Media error	Error Eje x	Error Eje y	Error Eje z	Porcentaje de convergencia	SEP	DRMS
Chan	2,46	1,78	0,56	0,74	100,00%	1,57	2,01
Levenberg	2,89	2,14	0,20	1,23	100,00%	1,82	2,48
Descent	2,54	1,45	0,89	0,45	100,00%	1,42	1,76
Taylor	2,68	1,65	0,79	0,68	54,55%	1,59	1,95
Taylor chan	2,41	1,78	0,54	0,81	72,73%	1,60	2,03
Friedlander	2,95	1,85	0,75	0,65	100,00%	1,66	2,10

Tabla 3.11: HVC 5 antenas

Algoritmo	Media error	Error Eje x	Error Eje y	Error Eje z	Porcentaje de convergencia	SEP	DRMS
Chan	1,17	0,58	0,44	0,79	100,00%	0,92	1,07
Levenberg	1,35	0,56	0,57	0,93	100,00%	1,05	1,23
Descent	0,45	0,17	0,27	0,27	100,00%	0,36	0,42
Taylor	1,32	0,84	0,32	0,58	61,54%	0,89	1,07
Taylor chan	0,82	0,37	0,46	0,68	100,00%	0,77	0,90
Friedlander	1,16	0,41	0,41	0,90	100,00%	0,88	1,07

Tabla 3.12: LTE 4 antenas

Algoritmo	Media error	Error Eje x	Error Eje y	Error Eje z	Porcentaje de convergencia	SEP	DRMS
Chan	1,07	0,28	0,32	0,94	100,00%	0,79	1,03
Levenberg	0,43	0,14	0,30	0,26	100,00%	0,36	0,42
Descent	0,41	0,17	0,33	0,16	100,00%	0,34	0,41
Taylor	0,41	0,09	0,22	0,31	92,31%	0,32	0,39
Taylor chan	0,73	0,24	0,31	0,55	100,00%	0,56	0,68
Friedlander	0,45	0,12	0,26	0,26	100,00%	0,33	0,39

Tabla 3.13: LTE 5 antenas

Algoritmo	Media error	Error Eje x	Error Eje y	Error Eje z	Porcentaje de convergencia	SEP	DRMS
Chan	1,30	0,59	0,46	0,86	100%	0,98	1,14
Levenberg	1,29	0,57	0,39	0,95	100%	0,97	1,17
Descent	0,83	0,43	0,33	0,50	100%	0,64	0,73
Taylor	1,13	0,42	0,42	0,88	100%	0,88	1,06
Taylor chan	1,02	0,38	0,44	0,75	100%	0,80	0,95
Friedlander	1,14	0,32	0,36	0,95	100%	0,83	1,06

Tabla 3.14: SB 4 antenas

Algoritmo	Media error	Error Eje x	Error Eje y	Error Eje z	Porcentaje de convergencia	SEP	DRMS
Chan	1,85	0,76	0,66	1,24	100%	1,36	1,60
Levenberg	2,49	1,09	1,48	1,32	100%	1,98	2,26
Descent	2,30	0,70	1,05	1,47	100%	1,64	1,94
Taylor	1,43	0,36	0,28	1,31	80%	1,00	1,39
Taylor chan	1,24	0,35	0,42	1,06	100%	0,93	1,19
Friedlander	2,22	0,43	0,35	2,10	100%	1,47	2,17

Tabla 3.15: SB 5 antenas

Las primeras conclusiones que se pueden extraer de las tablas anteriores son las siguientes:

- Los algoritmos de Chan y Friedlander obtienen peores resultados para los escenarios con 5 antenas.
- El algoritmo de Taylor posee el peor porcentaje de convergencia tal como se estudió en el apartado anterior.
- Los algoritmos de Chan y Friedlander obtienen peores resultados para los escenarios con 5 antenas.
- El algoritmo de Taylor-Chan mejora el porcentaje de convergencia del método de Taylor.
- La precisión de Taylor-Chan es superior o similar a la precisión del método de Taylor en condiciones normales.
- Para los escenarios de Carlos III y Jerónimo de Ayanz II utilizando 5 antenas, el algoritmo de Taylor-Chan es peor que el algoritmo de Taylor porque en esos casos el algoritmo de Chan tiene una precisión deficiente.
- Los algoritmos en general dependen bastante de la geometría del escenario.

Utilizando los resultados de las tablas anteriores, se pueden generar unos gráficos en los que se realiza una clasificación por escenario y así poder estudiar la influencia de la geometría del escenario con la precisión de los algoritmos.

Para poder realizar una comparación justa, es necesario conocer las dimensiones de los escenarios utilizados porque no es lo mismo tener un error de 1 metro en un escenario de 100m^2 que en uno de 10m^2 .

En la siguiente tabla se muestran las dimensiones aproximadas de los diferentes escenarios ordenadas de mayor a menor tamaño.

Escenario	Dimensiones		Área (m ²)
	Longitud (m)	Anchura (m)	
HVC	28	18	504
CARLOS III	11,2	5,2	60,64
JERONIMO DE AYANZ I	12,2	4,9	59,78
SB	6	5	30
LTE	7,8	3,8	29,64
JERONIMO DE AYANZ II	5,6	4	22,4

Tabla 3.16: Dimensiones de los escenarios

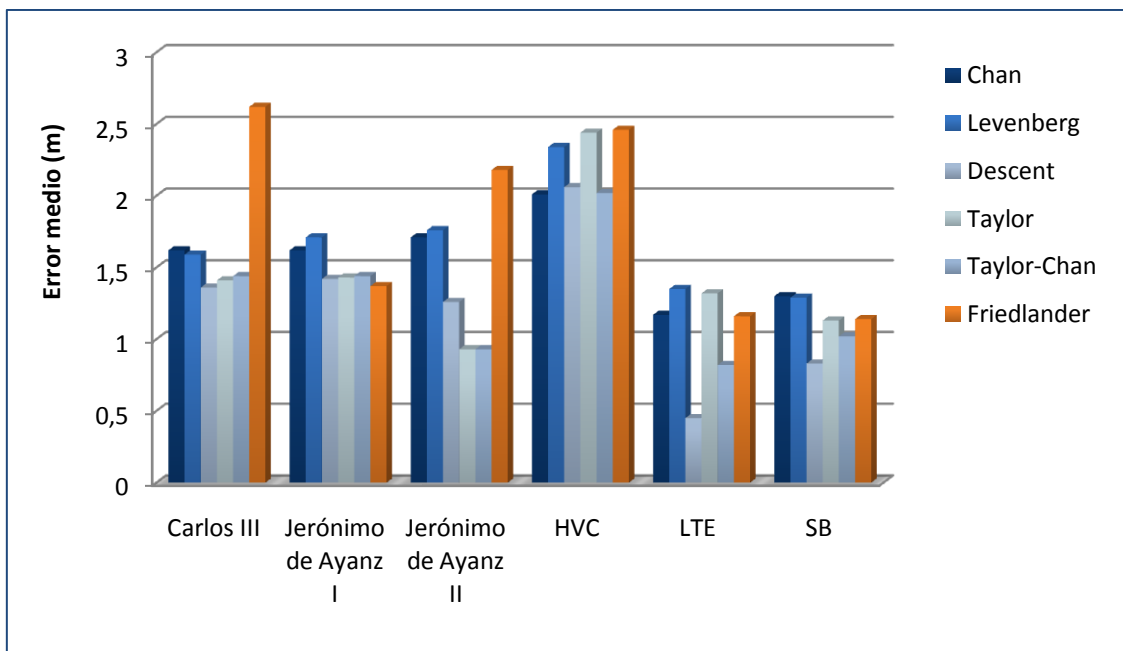


Figura 3.29: Error medio según el escenario utilizando 4 antenas

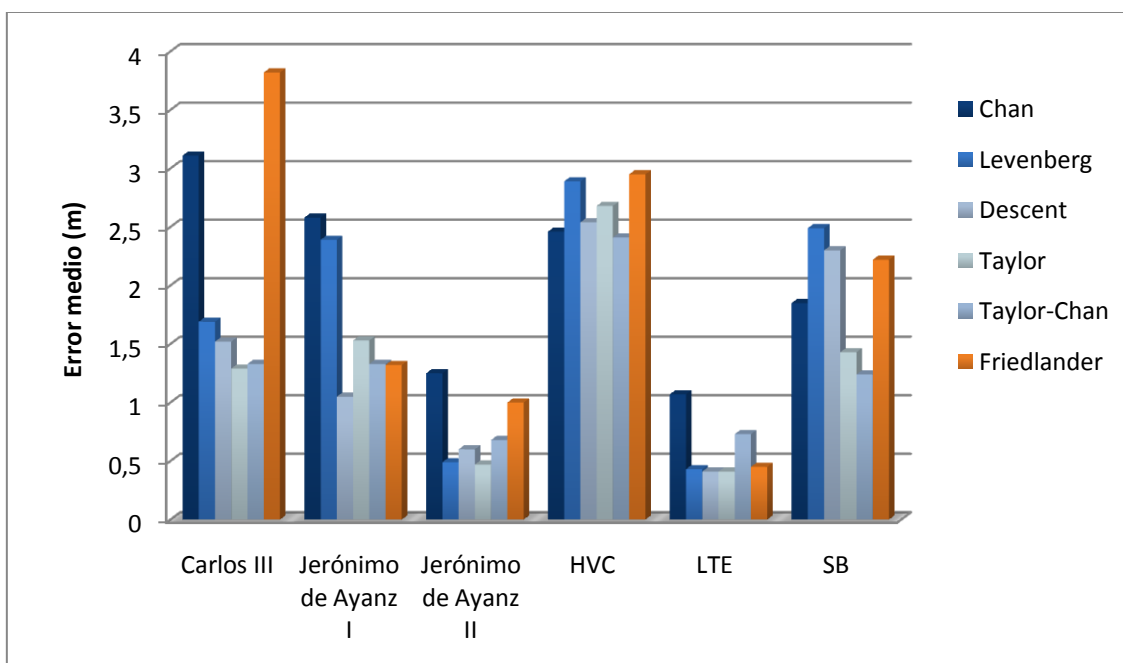


Figura 3.30: Error medio según el escenario utilizando 5 antenas

Si se observa la figura 3.29, el escenario que ofrece peores resultados es HVC con un error medio aproximado de 2 metros, seguido del escenario Carlos III y Jerónimo de Ayanz I. Si se comparan dichos resultados con la tabla 3.16 se comprueba fácilmente que los escenarios con mayor área ofrecen peores resultados.

Por lo tanto si nos fijamos en los escenarios más pequeños, LTE, SB y Jerónimo de Ayanz II, las precisiones aumentan. El error medio de estos tres escenarios oscila entre 0.4 y 1.30 metros para 4 antenas.

En el caso de la figura 3.30 ocurre algo similar aunque no existen tanta homogeneidad dentro de cada escenario como en la figura x1. Por un lado los escenarios LTE y Jerónimo de Ayanz II poseen las mejores precisiones alcanzando unas precisiones de aproximadamente 0.5m metros, mejorando los resultados obtenidos para 4 antenas, y por el otro lado en el escenario HVC se obtienen los mayores errores en torno a los 2.6 metros empeorando los resultados para el caso de 4 antenas.

Es necesario destacar que la precisión obtenida por los algoritmos Chan y Friedlander en escenarios grandes no es la esperada, tal como se observa en las figuras 3.29 y 3.30.

Para poder realizar una comparación total se ha procedido a calcular la media de los resultados mostrados en las figuras anteriores para obtener un valor único por algoritmo y número de antenas. Los resultados se muestran en la siguiente figura.

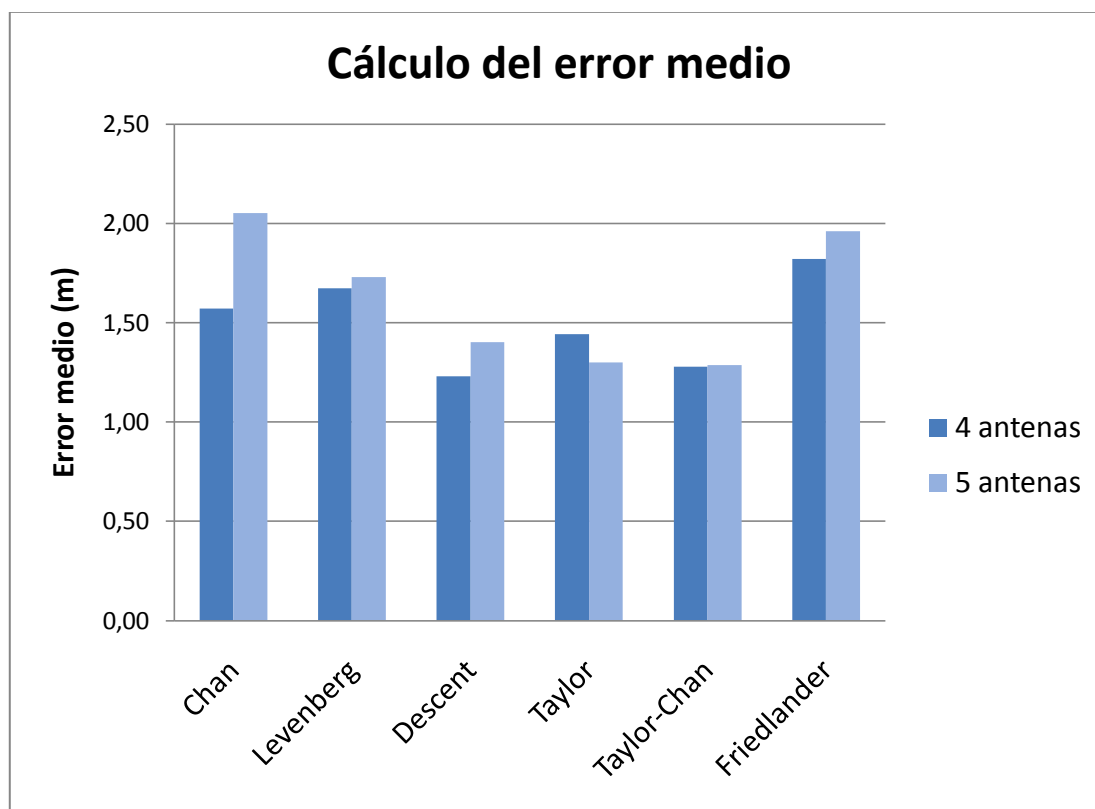


Figura 3.31: Errores medios por algoritmo

Según este gráfico, los algoritmos de Chan, Friedlander y Levenberg obtienen los peores resultados aunque en cualquier caso son inferiores a los 2 metros. En el caso de los dos primeros algoritmos esto se debe a su inadecuado comportamiento en los escenarios de mayor tamaño a pesar de obtener muy buenos resultados en los escenarios de tamaño medio o pequeño.

El algoritmo de Levenberg se comporta de manera homogénea en los diferentes escenarios pero en cualquier caso su precisión es ligeramente inferior al resto de los algoritmos estudiados.

Sobre el resto de los algoritmos es necesario destacar que los algoritmos de Taylor y Taylor-Chan alcanzan muy buenos resultados alcanzando precisiones de 1.25 metros pero hay que mencionar que no siempre convergen como se muestra en la tabla 3.17.

Convergencia	Taylor	Taylor-Chan
4 antenas	70,25%	84,97%
5 antenas	81,85%	88,17%

Tabla 3.17: Porcentajes de convergencia medios

Por lo tanto el algoritmo de Steepest Descent es el algoritmo con mejor precisión de los estudiados. Los resultados medios obtenidos con dicho algoritmo han sido 1.23 metros en el caso de 4 antenas y 1.40 metros para 5 antenas.

Por último se ha realizado un estudio de una posible aplicación de estos algoritmos. La aplicación pensada ha sido el seguimiento de un individuo a lo largo de su casa. A la hora de elegir los algoritmos a utilizar se ha observado la tabla 3.4 por ser la casuística de este estudio.

Se han desechado los algoritmos de Taylor, Taylor-Chan y Friedlander. Las razones para los dos primeros han sido los bajos porcentajes de convergencia y el algoritmo de Friedlander no ofrecía una buena precisión. Por lo tanto el estudio se ha realizado utilizando los algoritmos de Chan, Levenberg y Steepest Descent.

Se ha utilizado el escenario de Carlos III por ser una vivienda y se ha creado un recorrido a lo largo de la misma compuesto por 30 puntos. El recorrido ideado se muestra en la siguiente figura:

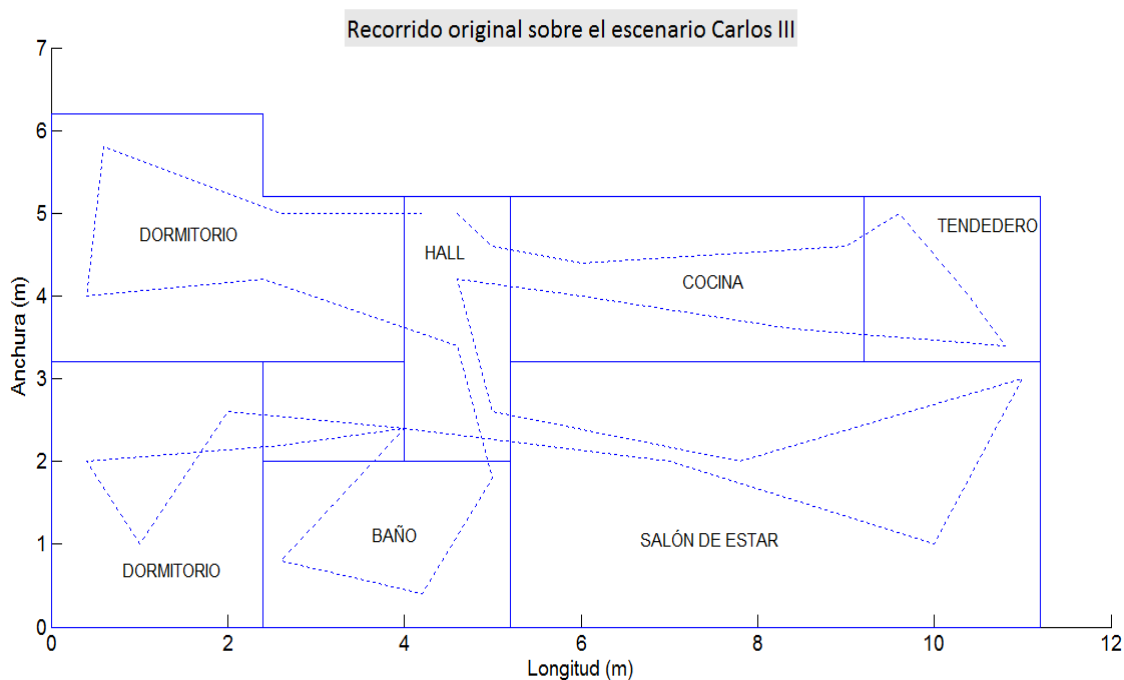


Figura 3.32: Recorrido sobre el plano del escenario Carlos III

Los resultados obtenidos numéricamente se muestran en la tabla 3.18. Se observan unos resultados similares a los obtenidos en los apartados anteriores.

Recorrido	Algoritmo		
	Chan	Levenberg	Descent
Error medio (m)	1,66	1,7	1,35

Tabla 3.18: Resultados obtenidos para el seguimiento de un individuo

Sin embargo, en este caso los errores medios no ofrecen una idea clara de la eficacia de los algoritmos. Por lo tanto, una solución gráfica ofrecería una visión más clara de los resultados. A continuación, se van a mostrar tres gráficas de la planta del escenario estudiado en las que se muestran los recorridos estimados por cada algoritmo.

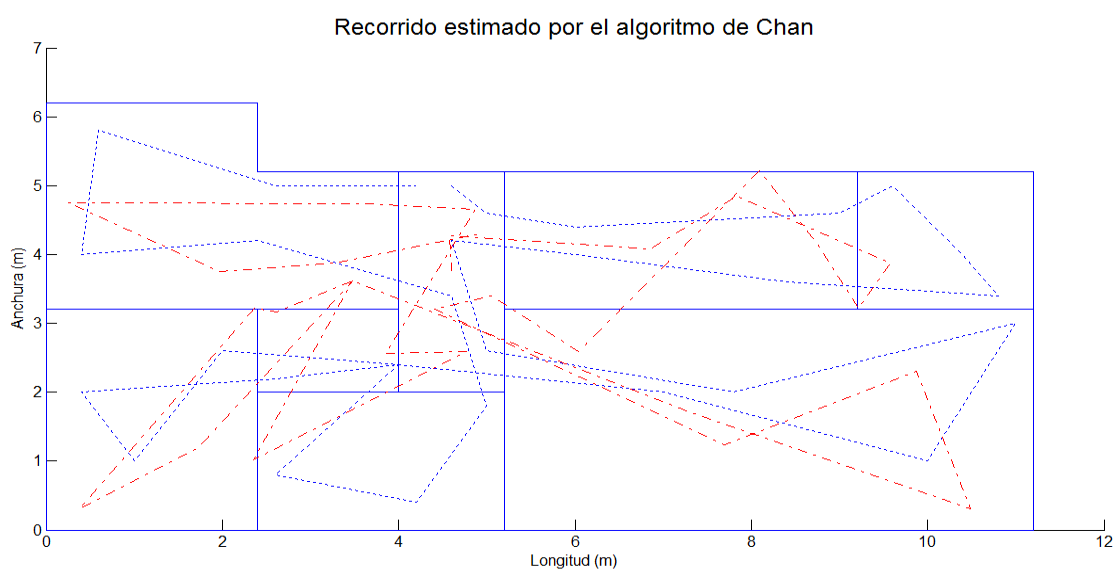


Figura 3.33: Recorrido obtenido con el algoritmo de Chan

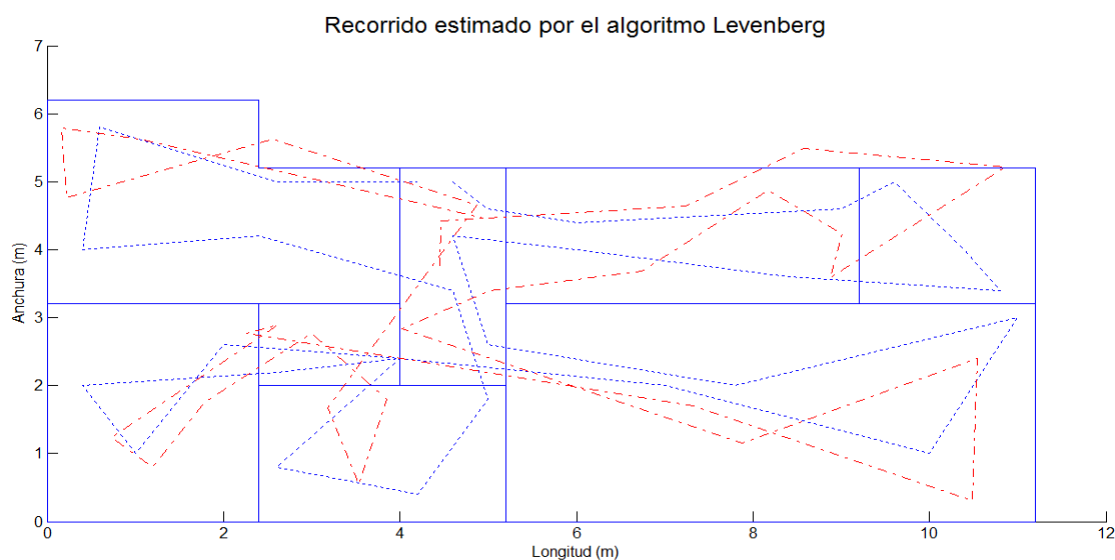


Figura 3.34: Recorrido obtenido con el algoritmo de Levenberg

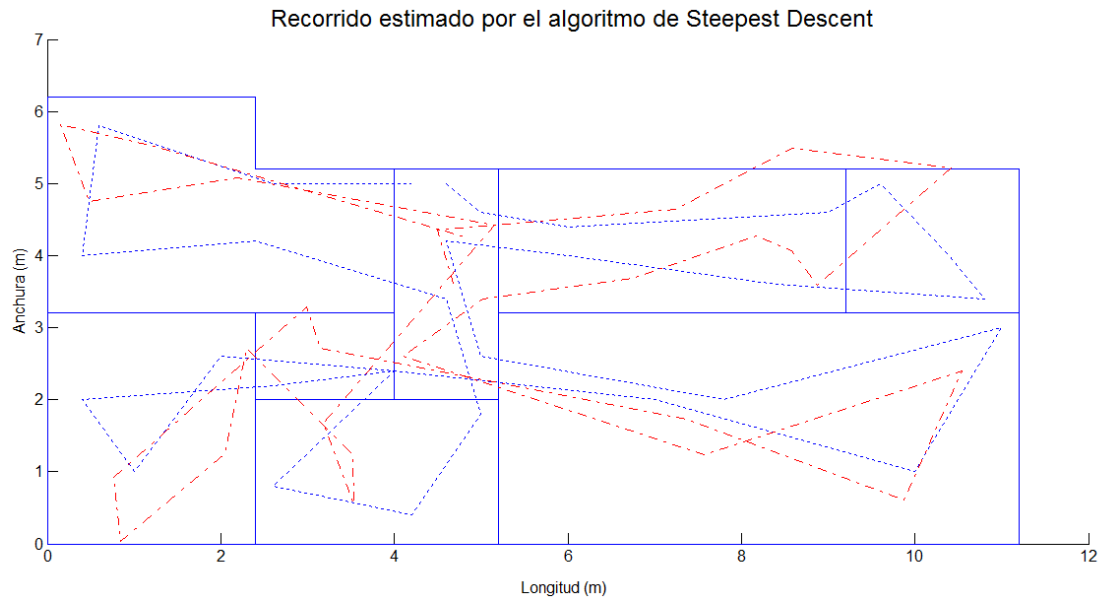


Figura 3.35: Recorrido obtenido con el algoritmo de Steepest Descent

Tal como se observa en las tres figuras anteriores, los recorridos estimados se acercan bastante al recorrido original. Existen puntos donde aparece un error lo suficientemente alto como para estimar puntos fuera del escenario pero ese problema es fácilmente solucionable.

A pesar de mostrar las figuras en 2D, la solución es totalmente válida para 3D. Sin embargo para la aplicación en cuestión no es necesario utilizar la tercera componente estimada.

3.3.2.2.2. RSS

Por último se van a mostrar los resultados de la técnica KNN basada en potencias y tal como se ha descrito anteriormente se va a utilizar el método del vecino más cercano ($K = 1$).

Se ha elegido un escenario de tamaño medio por la ardua tarea del entrenamiento previo necesario: Jerónimo de Ayaz I. A pesar del tamaño del escenario, para conseguir una resolución media de 1 metro es necesario discretizar el escenario con 180 puntos y por tanto el mismo número de simulaciones. Los parámetros que se han utilizado se muestran en la tabla 3.19.

Parámetro	Valor
Resolución	1 metro
Potencia transmitida	1 mW
Rebotes	6
Frecuencia	2.4 GHz

Tabla 3.19: Parámetros de simulación utilizados para KNN

Se han supuesto dos casos en el escenario; uno con 4 antenas y otro con 5 antenas. Para cada uno de ellos se ha calculado el error medio de las estimaciones y el mínimo error que se obtendría si el algoritmo estimara siempre el vecino más cercano. En la siguiente tabla se muestran los resultados obtenidos:

Escenario		Error medio (m)	Mínimo error posible (m)
Jerónimo de Ayanz I	4 antenas	2,36	0,64
	5 antenas	2,72	0,64

Tabla 3.20: Resultados KNN

Si recordamos los datos obtenidos con TDOA en el mismo escenario obtenemos la siguiente gráfica:

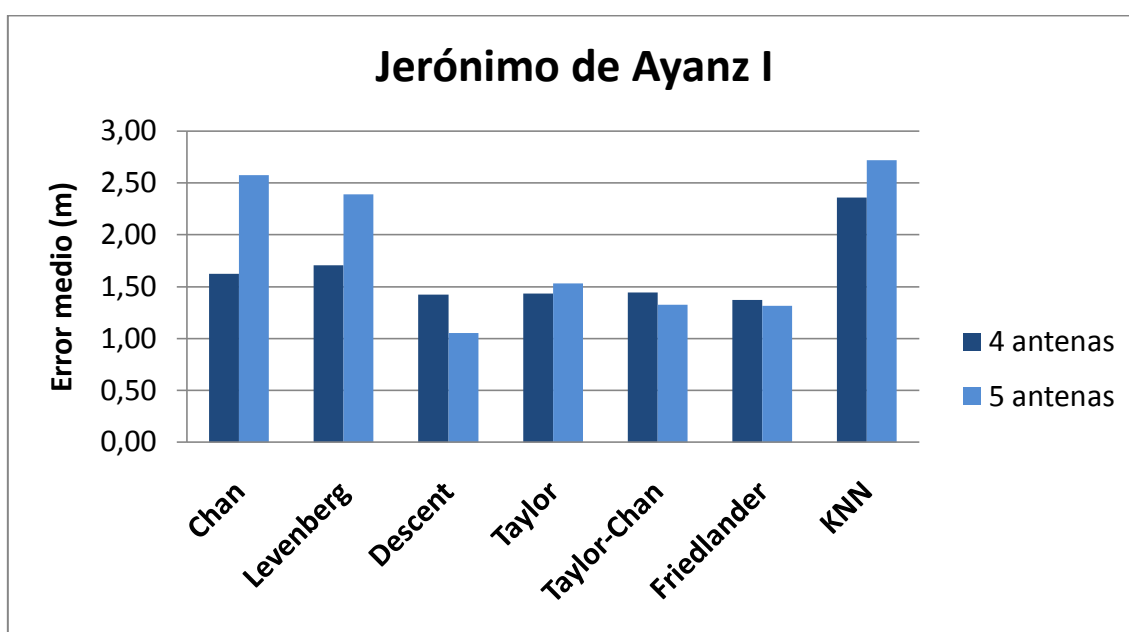


Figura 3.36: Comparación de resultados de TDOA y RSS

Como se observa en la figura 3.36, el algoritmo de KNN es el menos preciso de todos para las dos casuísticas estudiadas en cuanto al número de antenas. La razón principal es la resolución elegida para el escenario ya que realizando un entrenamiento más preciso el error disminuiría. Por otro lado, la ventaja clara es el tiempo de cómputo y la característica de no tener la opción de suministrar coordenadas que no se hayan definido con anterioridad.

Una mejora de este algoritmo, comentada en apartados anteriores, es referenciar el cálculo de la posición $n+1$ con la posición n . Esta mejora es aplicable en cálculos de recorridos de objetos o personas en tiempo real imponiendo un criterio lógico en cuanto a la distancia máxima recorrida entre dos puntos estimados consecutivos.

De esta forma a la hora de calcular posiciones se tienen dos referencias, ya que se calculan los vecinos más cercanos según las potencias de llegada pero se elige el más cercano al punto estimado anterior.

Memoria

Proyecto real

Bloque 4

4. Proyecto real

4.1. PROYECTO REAL IECISA

El proyecto que ya se ha mostrado en los capítulos anteriores nació dentro de una beca en la empresa Informática El Corte Inglés (IECISA) en Madrid. La beca comenzó en Octubre y ha sido desarrollada en el departamento de Ingeniería. Al comienzo de la misma, se presentó la posibilidad de realizar un proyecto centrado en los sistema de localización la cual motivó el tema de este proyecto.

El proyecto en cuestión pretendía utilizar la técnica de localización TDOA para la localización de terminales móviles en el ámbito de los sistemas de emergencia 112. Antes de nada era necesario conocer qué era TDOA y por ello ahí comenzó un estudio profundo de esta técnica.

La actuación de IECISA pasaba por entender el sistema de localización a implantar realizando un estudio de consultoría y posteriormente implantarlo en la región de interés para comprobar su correcto funcionamiento. La idea principal era obtener el conocimiento de su instalación para realizar la integración en futuros proyectos en otros sistemas 112 españoles.

El sistema de localización comercial elegido pertenece a la empresa estadounidense True Position y está fuertemente implantado en Estados Unidos. El sistema estudiado utiliza U-TDOA (Uplink Time Difference of Arrival) con las llamadas entrantes para estimar su posición.

Se basa en implantar hardware en paralelo con la red móvil de las operadoras para poder estimar las posiciones de los usuarios. Los equipos hardware que se requieren se muestran en la figura de la página siguiente. El equipamiento de la figura se describe brevemente a continuación:

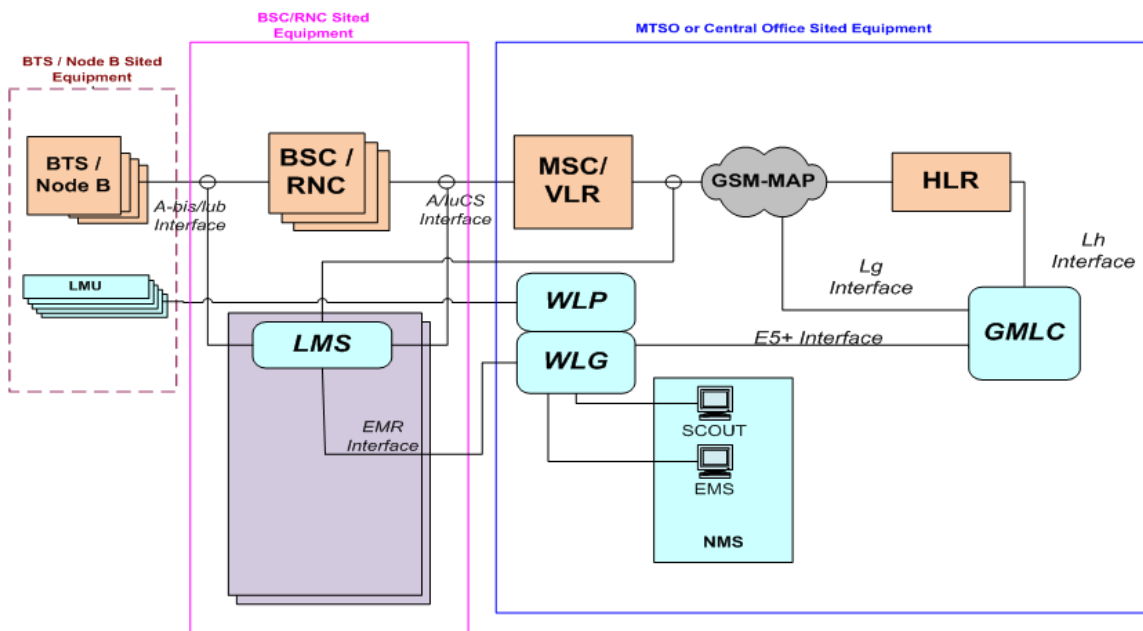
- **LMU (Location Measurement Unit)**

Recolecta las medidas de tiempos de llegada utilizadas en los cálculos TDOA. Se instala uno en cada BTS y soporta desde 700MHz hasta 3.9GHz. Actualmente soporta GSM, UMTS, HSPA, CDMA, CDMA 2000, WiMAX y LTE.

- **WLP (Wireless Location Processor)**

Recolecta la información de tiempos obtenidos por las LMUs. Se instala uno por cada BSC. Es capaz de implementar también AOA o un algoritmo híbrido para mejorar las estimaciones de las localizaciones. A su vez cada servidor WLP es capaz de soportar 672 LMUs.

- **WLG (Wireless Location Gateway)**
Acciona el TPLP (True Position Location Platform) para calcular la ubicación y distribuirla a la entidad solicitante. Es un sistema que debe ser redundado en cada plataforma de True Position.
- **NMS (Network Management System)**
Se utilizan dos sistemas de gestión de red como son SCOUT (System Configuration, Optimization and Utilities Tool) y EMS (Element Management System)
- **LMS (Link Monitoring System)**
Utilizado para proveer localizaciones a sistemas propietarios. Permite al sistema operar de manera independiente a la red móvil y actuar por medio de alarmas en cuanto al origen de las llamadas o los números marcados.
- **GMLC (Gateway Mobile Location Center)**
Es un gateway de aplicación para el equipamiento de localización. Permite integrar el servicio de localización con los sistemas de emergencias.



- Equipamiento de la operadora
- Plataforma Localización TruePosition

Arquitectura del sistema de localización de True Position

Tras conocer la arquitectura de la solución, se va a mostrar la solución implantada en la región de Murcia para comprobar su funcionamiento. En concreto se eligió la zona de La Manga del Mar Menor para su implantación.

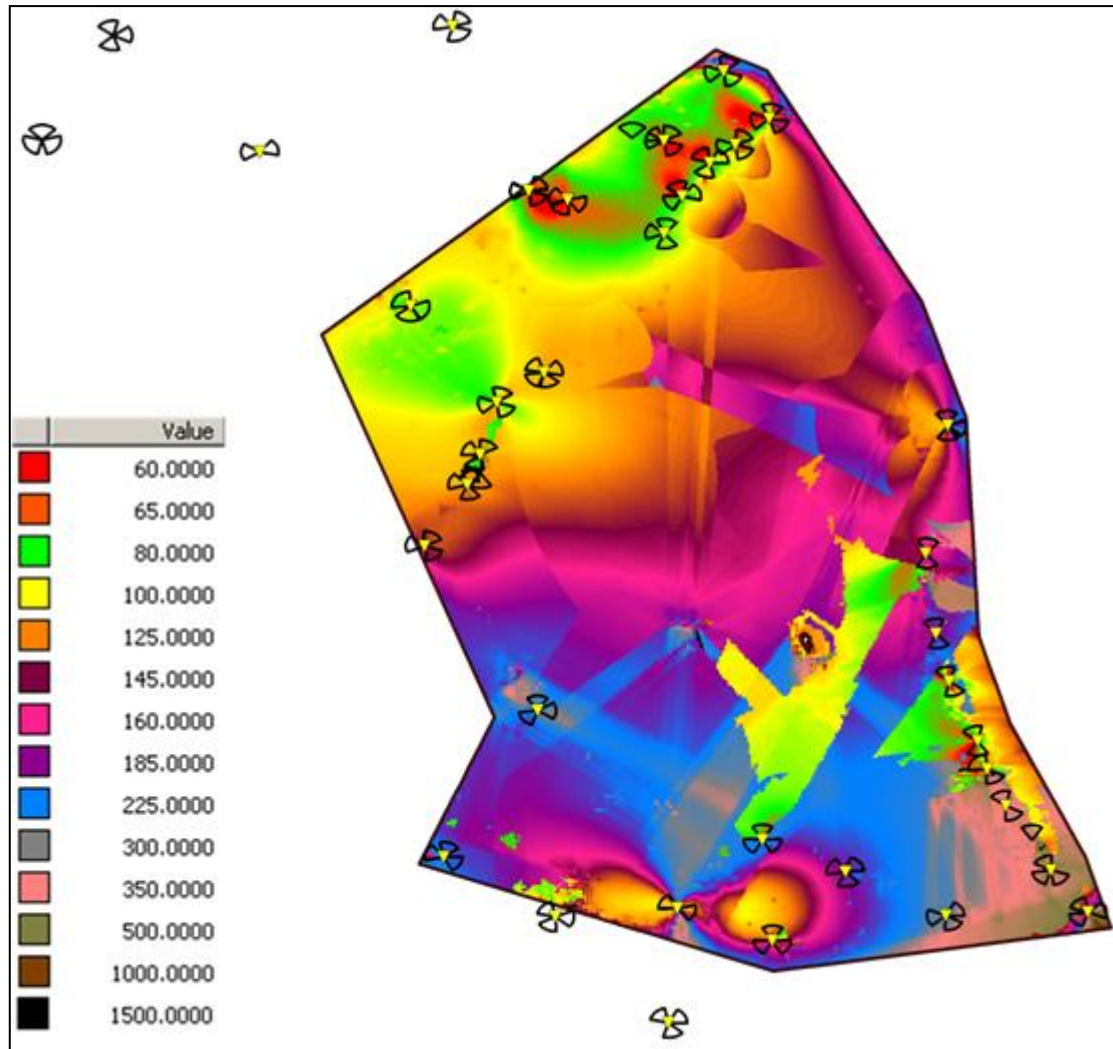


Localización de los sites y las zonas del proyecto

100

fase posterior extenderla a la red de Vodafone. Las gráficas mostradas a partir de ahora pertenecen a estudios basados en estaciones base de Orange.

Las simulaciones llevadas a cabo utilizan tanto la banda de 900 como la de 1800MHz para obtener una precisión combinada de ambas bandas y el resultado se muestra en la siguiente figura donde las unidades de la leyenda son metros.



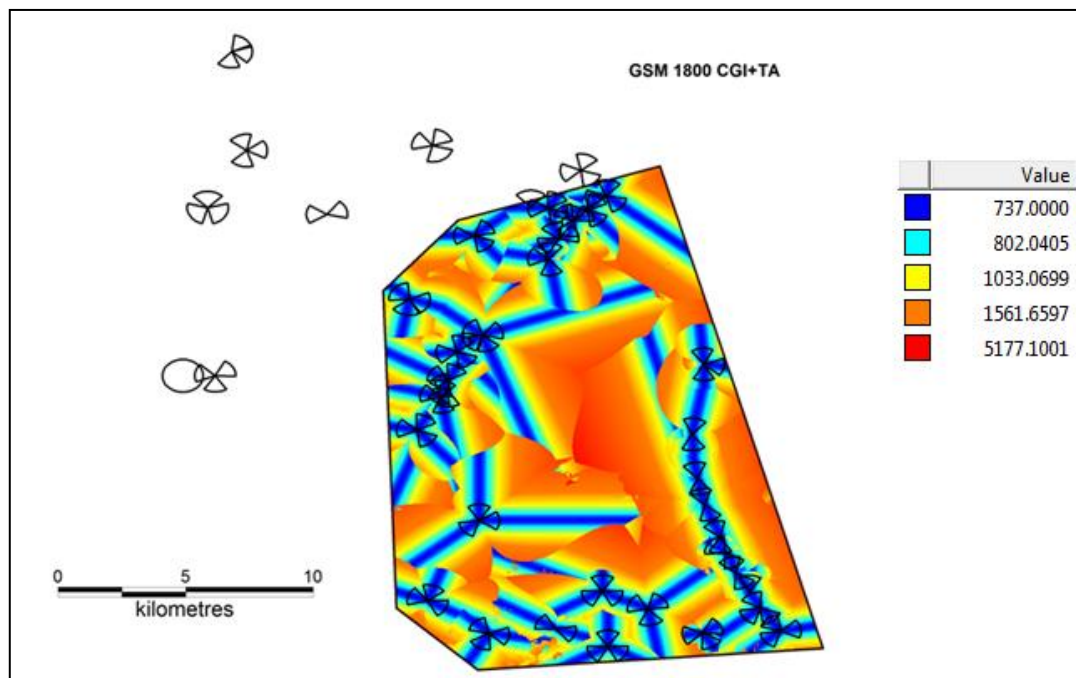
Previsiones de precisión en La Manga del Mar Menor

Se observa que en las zonas con una alta densidad alta de estaciones base, como puede ser el polígono norte, se obtienen los mejores resultados alcanzando una precisión promedio de 96.3 metros.

Por otro lado las peores precisiones obviamente se encuentran el mar donde los efectos de la reflexión se agravan y por tanto es más difícil estimar la posición real del terminal móvil.

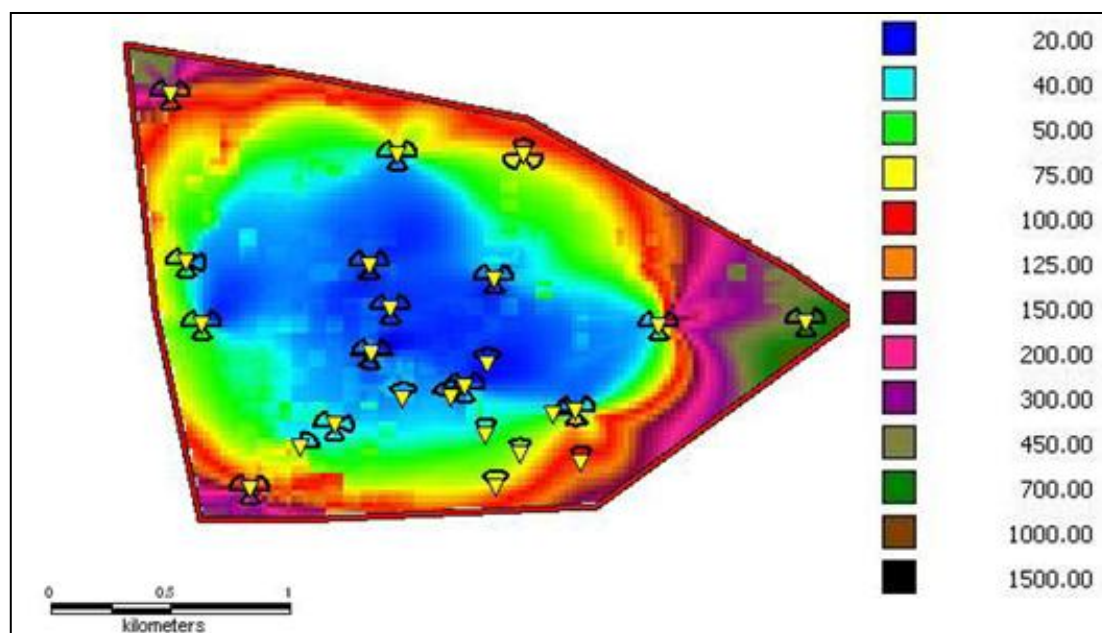
En el estudio llevado a cabo se compararon los resultados con otra técnica utilizada actualmente como CellID que se basa en localización por celda de estación

base. Con ese método la precisión alcanzada en la región total es muy inferior tal y como se muestra en la siguiente figura. La leyenda también tiene unidades de metros.



Precisión utilizando la técnica CellID en la Manga del Mar Menor

La última parte del estudio se centró en la simulación del sistema en el centro de Murcia donde la densidad de estaciones base es muy grande. En este caso la precisión es muy alta tal como se muestra en la figura siguiente:



Precisión en el centro de Murcia

A continuación se verá que esta herramienta de simulación es muy previsor y que debe ser mejorada ya que se aleja bastante de la realidad.

Tras comprobar la mejora teórica de los resultados frente a cualquier sistema existente hoy en día de localización de terminales móviles, se decidió realizar la prueba piloto en la zona estudiada. Actualmente ya se ha realizado la fase de implantación en la red de Orange y su integración con el sistema de despacho en el 112 de Murcia y se está estudiando cómo realizar la integración con la red de Vodafone.

Las pruebas llevadas a cabo desde entonces han dado lugar a unos resultados mucho mejores de lo esperado. De las tres zonas mostradas en una figura anterior, ya se comentó que los peores resultados teóricos se daban en la zona del mar por lo tanto esta zona ha sido la más estudiada para evaluar la viabilidad del proyecto.

Las pruebas han tenido dos fases. En la primera de ellas se realizaron experimentos en el polígono norte y en el sur y en la segunda se realizaron en el mar por medio de un barco. Las conclusiones a las que se han llegado son las siguientes:

- **Los resultados en la zona del mar tienen un margen de error que se encuentra entre 0.1 y 20 metros.**
- **La precisión en los polígonos definidos es siempre menor a 10 metros.**
- **Se ha conseguido reducir en un 25% el número total de estaciones base a utilizar, definido en el inicio del proyecto, porque aun así se consiguen unas precisiones muy altas.**
- **A día de hoy, los resultados obtenidos han superado cualquier sistema existente y se está estudiando su incorporación en otros sitios.**

En definitiva, este proyecto actualmente en ejecución está siendo todo un éxito y puede dar lugar a grandes avances en el mundo de los sistemas de emergencia ayudando a salvar vidas. Por lo tanto, los sistemas de localización aún tienen mucho que decir en el ámbito civil.

Memoria

Conclusiones

Bloque 5

5. Conclusiones y líneas futuras

5.1. CONCLUSIONES

En este apartado se van a mostrar las conclusiones obtenidas a través de los estudios y simulaciones mostradas en bloque anteriores.

En la primera parte del bloque 3 se ha realizado un estudio teórico de los sistemas de localización que culmina con una tabla resumen, tabla 3.1. Las primeras observaciones a mencionar son los problemas observados en los algoritmos de Fang e interpolación esférica. Dichos algoritmos se descartan por el hecho de poseer requisitos en cuanto al número de medidas necesarias para su funcionamiento; 4 medidas en el primer caso y un mínimo de 5 en el segundo.

Otro aspecto importante de dicho estudio ha sido comprobar los requisitos en cuanto a estimaciones iniciales como entrada a ciertos algoritmos.

El siguiente paso fue realizar el estudio práctico. En dicho estudio ya sólo se implementaron los algoritmos que a priori eran más robustos. Las conclusiones resultantes son las siguientes:

- Dos de los algoritmos con requisitos de posición inicial (Taylor y Levenberg) poseen una buena precisión pero el hecho de poseer dicho requisito la geometría y los elementos del escenario les afecta en un alto grado.
- El algoritmo de Taylor lo desestimamos para implementaciones futuras por sus problemas de convergencia. Tal como se ha explicado y demostrado dentro del apartado 3.3.2.2.1 se pueden implementar mejoras para aumentar el porcentaje de convergencia para este algoritmo pero en ningún caso conseguimos un porcentaje del 100%.
- Los algoritmos de Chan y Friedlander sufren peor los problemas del multitrayecto y por ello sus precisiones son ligeramente inferiores a las conseguidas con los demás algoritmos.
- El algoritmo de Steepest Descent ha mostrado los mejores resultados obteniendo una precisión media de 1.3 metros en los escenarios simulados. Tras el estudio se comprueba su robustez ante cualquier tipo de escenario a pesar de requerir una posición inicial para su funcionamiento.

En definitiva, el algoritmo de Steepest Descent sería el método elegido para cualquier tipo de proyecto basado en localización indoor si nos acogemos a los resultados de este proyecto. Un ejemplo de ello es la aplicación de seguimiento mostrada en el bloque 3 que refleja un alto grado de precisión con los algoritmos empleados consiguiendo con el algoritmo de Steepest Descent los mejores resultados

Por otro lado, la técnica RSS ha demostrado ser ligeramente peor que la técnica TDOA aunque este resultado es engañoso porque en el caso de realizar un entrenamiento previo muy preciso con una resolución muy alta, se conseguirían resultados similares o mejores que los obtenidos con TDOA.

En resumen, dependiendo de la aplicación de interés para el usuario, se elegiría un algoritmo u otro. Es obvio pensar que eligiendo una tecnología como Zigbee con decenas de tags a lo largo del escenario se alcanzan mejores resultados que utilizando solamente cuatro puntos de acceso WiFi por mucho que se empleen los mejores algoritmos por detrás. Incluso una solución mixta que utilice AOA y TDOA obtendrá mejores resultados con la implicación del coste por supuesto.

En conclusión, el compromiso precisión frente a coste es el factor más determinante en el campo de la localización indoor. El conocimiento teórico y matemático obtenido con este proyecto nos provee de un criterio a la hora de valorar las soluciones comerciales existentes y poder decidir la mejor técnica para cada aplicación.

5.2. LÍNEAS FUTURAS

A lo largo de este proyecto se ha realizado un análisis de las diferentes técnicas de localización indoor. El proyecto se ha centrado en escenarios más o menos reales, en los que existen pocas antenas disponibles para el target, sin embargo se puede realizar un estudio más avanzado con las siguientes líneas de estudio para comprobar la coherencia de los resultados:

- Evaluación de los diferentes algoritmos al aumentar el número de antenas disponibles (6, 7, 8,...) considerando escenarios tipo, existentes en la universidad, como la biblioteca o el edificio del departamento.
- A raíz de la anterior línea, un estudio con más detalle del algoritmo de Chan en el que se distingue la casuística de 4 antenas frente a un número mayor de ellas.

Además los resultados obtenidos gracias al desarrollo de la herramienta de trazado de rayos podrían tener una mayor utilidad siguiendo las siguientes líneas de acción:

- Pruebas reales de toma de medidas en los escenarios evaluados en este proyecto.
- Diseño e implantación de una prueba piloto de localización utilizando TDOA en algún edificio utilizando por ejemplo la red WiFi existente.

Las técnicas estudiadas y comparadas están en pleno desarrollo comercial por lo que encontrar un sistema fiable y preciso podría ser de vital importancia hoy en día. Tal como se comentó en bloque 2 de este proyecto la localización en entornos domóticos podría tener gran utilidad por lo que una integración entre un sistema de domótica generado con un sistema de localización podría ser un campo de desarrollo en la UPNA.

En definitiva, los sistemas de localización tienen múltiples aplicaciones hoy en día por lo que una vez conocidas las técnicas existentes y las precisiones estimadas alcanzadas por cada una de ellas, el siguiente paso lógico consistiría en encontrar un ámbito de aplicación interesante para aplicar los conocimientos adquiridos.

Memoria

Anexos

Bloque 6

6. ANEXOS

6.1. BIBLIOGRAFÍA

- [1] [2] Guoqiang Mao and Baris Fidan, “Localization algorithms and Strategies for Wireless Sensor Networks”, Premier Reference Source, 2009.
- [3] W. Foy, “Position-Location Solutions by Taylor-Series Estimation”, *IEEE Trans. Aerospace and Elect. Sys.*, vol. AES-12, Mar. 1976, pp. 187–93.
- [4] B. Fang, “Simple Solutions for Hyperbolic and Related Position Fixes”, *IEEE Trans. Aerospace and Elect. Sys.*, Sept. 1990, pp. 734–53.
- [5] J. Smith and J. Abel, “The Spherical Interpolation Method of Source localization”, *IEEE J. Oceanic Eng.*, vol. OE-12, Jan. 1987, pp. 246–52.
- [6] H. C. Schau and A. Z. Robinson, “Pasive source localization employing intersecting spherical surfaces from time-of-arrival differences” *IEEE Trans. Acoust, Speech Signal Processing*, vol. ASSP-35, PP 1223-1225, Aug. 1987.
- [7] K. Ho and Y. Chan, “Solution and Performance Analysis of Geolocation by TDOA,” *IEEE Trans. Aerospace and Elect. Sys.*, vol. 29, Oct. 1993, pp. 1311–22.
- [8] Y.T. Chan and K.C. Ho, “A simple and *efficient* estimator for *hyperbolic* location”, *IEEE Trans. Signal Process.* 42 (8) (August 1994), pp. 1905–1915.
- [9] B. Friedlander, “A Passive Localization Algorithm and Its Accuracy Analysis”, *IEEE Journal of Oceanic Engineering*, vol. OE-12, no. 1, pp. 234{244, January 1987.
- [10] J. S. Abel, “A Divide and Conquer Approach to Least-Squares Estimation”, *IEEE Transactions on Aerospace and Electronic Systems*, vol. 26, pp. 423{427, March 1990.
- [11] Lingwen Zhang and Zhenhui Tan, “A new TDOA algorithm based on Taylor series expansion in cellular networks”, *Frontiers of Electrical and Electronic Engineering in China*, vol3, number 1, January 2008.

- [12] Fermín Esparza, “Algoritmo de trazado de rayos en tres dimensiones para la caracterización de entornos indoor”, Proyecto Final de Carrera, Universidad Pública de Navarra, 2009.
- [13] Fredrik Gustafsson and Fredrik Gunnarsson, “Positioning using time-difference of arrival measurements”, *ICASSP*, Hongkong, PRC, 2003.
- [14] Alan Bensky, “Wireless Positioning. Technologies and applications”, GNSS Technology and applications series, Artech House, 2008.
- [15] J. J. Astrain y J. Villadongos y A. Córdoba, “Localización y seguimiento de terminales móviles en entornos indoor”, Departamento de Matemática e Informática, Universidad Pública de Navarra.
- [16] A. R. Jimenez, F. Seco, C. Prieto y J. Roa, “Tecnologías sensoriales de localización para entornos inteligentes” Instituto de Automática Industrial (CSIC), 2005.

6.2. Código MATLAB

ALGORITMO DE CHAN

function posicion_estimada=estimacion_chan(tiempos_medidos, puntos_medidos)

N=length(tiempos_medidos);

%Es necesario distinguir entre 4 medidas y más de 4 medidas

if N==4

COL1=zeros(3,1);

COL2=zeros(3,1);

COL3=zeros(3,1);

for i=1:N-1

COL1(i,1)=puntos_medidos(i+1,1)-puntos_medidos(1,1);

COL2(i,1)=puntos_medidos(i+1,2)-puntos_medidos(1,2);

COL3(i,1)=puntos_medidos(i+1,3)-puntos_medidos(1,3);

end

A=-inv([COL1,COL2,COL3]);

K=zeros(4);

for i=1:N

K(i)=puntos_medidos(i,1)^2+puntos_medidos(i,2)^2+puntos_medidos(i,3)^2;

end

R=zeros(3,1);

for i=1:N-1

R(i,1)=tiempos_medidos(i+1)-tiempos_medidos(1);

end

B=zeros(3,1);

B(1,1)=(1/2)*(R(1)^2-K(2)+K(1));

B(2,1)=(1/2)*(R(2)^2-K(3)+K(1));

B(3,1)=(1/2)*(R(3)^2-K(4)+K(1));

E=A*R;

F=A*B;

a=1-E'*E;

b=2*(puntos_medidos(1,:)*E - F'*E);

c=2*puntos_medidos(1,:)*F - F'*F - K(1);

RAIZ=(-b+sqrt(b^2-4*a*c))/(2*a);

```

posicion_estimada=A*(R*real(RAIZ)+B);

elseif N>4

% primero matriz Q

Q=ones(N-1,N-1);
Q=Q/2;
for i=1:N-1
    Q(i,i)=1;
end

%Sigo con el resto de matrices

G=zeros(N-1,4);

for i=1:N-1
    G(i,1)=puntos_medidos(i+1,1)-puntos_medidos(1,1);
    G(i,2)=puntos_medidos(i+1,2)-puntos_medidos(1,2);
    G(i,3)=puntos_medidos(i+1,3)-puntos_medidos(1,3);
    G(i,4)=tiempos_medidos(i+1)-tiempos_medidos(1);
End

G=-G;

%calculo los k

K=zeros(N);

for i=1:N
    K(i)=puntos_medidos(i,1)^2+puntos_medidos(i,2)^2+puntos_medidos(i,3)^2;
end

h=zeros(N-1,1);

for i=1:N-1
    h(i,1)=(1/2) * ( (tiempos_medidos(i+1)-tiempos_medidos(1))^2 -K(i+1)+K(1) );
end

solucion= inv(G'*inv(Q)*G) *G'*inv(Q)*h;
posicion_estimada=[abs(solucion(1)),abs(solucion(2)),abs(solucion(3))];

%segunda iteracion

B=zeros(N-1, N-1);

for i=1:N-1
    B(i,i)=tiempos_medidos(i+1)-tiempos_medidos(1)+solucion(4);
end

```

```

Y=(3e8)^2*B*Q*B;
solucion2= inv(G'*inv(Y)*G) *G'*inv(Y)*h;
posicion_estimada2=[solucion2(1),solucion2(2),solucion2(3)];
else

    disp('Algoritmo de chan no existente')
    close all

end
end

```

ALGORITMO DE FANG

function [posicion] = estimacion_fang(x2,x3,x4,y3,y4,x,y,z)

```

xp=x;
yp=y;
zp=z;

R1=sqrt( xp^2 + yp^2 + zp^2);
R2=sqrt( (xp-x2)^2 + yp^2 + zp^2);
R3=sqrt( (xp-x3)^2 + (yp-y3)^2 + zp^2);
R4=sqrt( (xp-x4)^2 + (yp-y4)^2 + zp^2);

posicion=[];

aux=0;
for j=.000001:.0000005:.00001;
    aux=aux+1

    barra = waitbar(0,'Espere... ');

    for i=1:100

        R21=(R2-R1)*(1+j*0.1*randn);
        R31=(R3-R1)*(1+j*randn);
        R41=(R4-R1)*(1+j*randn);

        g1=(R31*x2/R21-x3)/y3;
        g2=(R41*x2/R21-x4)/y4;
        g=g1-g2;
        h1=(x3^2+y3^2-R31^2+R31*R21*(1-(x2/R21)^2))/(2*y3);
        h2=(x4^2+y4^2-R41^2+R41*R21*(1-(x2/R21)^2))/(2*y4);
        h=h1-h2;
        d1=-((1-(x2/R21)^2)+g1*g1);
        d2=-((1-(x2/R21)^2)+g2*g2);
        d=d1-d2;
        e1=x2*((1-(x2/R21)^2))-2*g1*h1;
    end
end

```



```

e2=x2*((1-(x2/R21)^2))-2*g2*h2;
e=e1-e2;
f1=((R21^2)/4)*(1-(x2/R21)^2)-h1*h1;
f2=((R21^2)/4)*(1-(x2/R21)^2)-h2*h2;
f=f1-f2;

x(i)=(-e+sqrt(e*e-4*d*f))/(2*d);
y(i)=g1*x(i)+h1;
z(i)=sqrt(d1*x(i)*x(i)+e1*x(i)+f1);
posicion=[posicion;[x(i),y(i),z(i)]];
xerr(i)= abs(xp-x(i));
yerr(i)= abs(yp-x(i));
zerr(i)= abs(zp-x(i));
err(i)=1000*sqrt((x(i)-xp)^2 + (y(i)-yp)^2 + (z(i)-zp)^2);

waitbar(i /100)

end
close(barra)

R21p=(R2-R1)*j*1000*39.37;
R31p=(R3-R1)*j*1000*39.37;
R41p=(R4-R1)*j*1000*39.37;

if R21p<R31p && R21p<R41p;
    RMS=sqrt((R21p)^2);
elseif R31p<R21p && R31p<R41p
    RMS=sqrt((R31p)^2);
else
    RMS=sqrt((R41p)^2);
end

%fprintf('Errores TDOA introducidos (m) = %g\n',j);
%fprintf('RMS errores en las medidas TDOA = %g\n',RMS);
%fprintf('Error medio en x = %g\n',mean(xerr));
%fprintf('Error medio en y = %g\n',mean(yerr));
%fprintf('Error medio en z = %g\n',mean(zerr));
%fprintf('Error medio total = %g\n',mean(err));
%fprintf('\n');

if j==.000001;
    nn=RMS;xx=mean(xerr);yy=mean(yerr);zz=mean(zerr);ee=mean(err);
else
    nn=[nn RMS];xx=[xx mean(xerr)];yy=[yy mean(yerr)];zz=[zz mean(zerr)];ee=[ee
mean(err)];
end

end

figure(1);

```

```

plot(nn,xx);grid;
title('Error en x vs Error medida RMS');
xlabel('Error medida RMS');
ylabel('Error en x');

figure(2);
plot(nn,yy);grid;
title('Error en y vs Error medida RMS');
xlabel('Error medida RMS');
ylabel('Error en y');

figure(3);
plot(nn,zz);grid;
title('Error en z vs Error medida RMS');
xlabel('Error medida RMS');
ylabel('Error en z ');

figure(4);
plot(nn,ee);grid;
title('Error total vs Error medida RMS');
xlabel('Error medida RMS ');
ylabel('Error total ');

end

```

ALGORITMO DE FRIEDLANDER

function [posición_est] = estimacion_friedlander (distancias_medidas,sensores)

```
N=length(distancias_medidas);
```

%Primero calculo las diferencias de distancias

```

r=zeros(N-1,1);
for i=1:N-1
    r(i,1)=distancias_medidas(i+1)-distancias_medidas(1);
end

```

*% Voy a calcular las diferentes matrices necesarias para calcular la
%solucion*

%Empiezo con Sj

```
S=zeros(N-1,3);
```

```
for i=1:N-1
```

```
    S(i,1)= sensores(i+1,1) - sensores(1,1);
```

```

        S(i,2)= sensores(i+1,2) - sensores(1,2);
        S(i,3)= sensores(i+1,3) - sensores(1,3);
    end

    %Sigo con uj

    u=zeros(N-1,1);
    R=zeros(N);

    for i=1:N
        R(i)=sqrt( sensores(i,1)^2 + sensores(i,2)^2 + sensores(i,3)^2 );
    end

    for i=1:N-1

        u(i,1)= 1/2 * ( R(i+1)^2 - R(1)^2 - r(i)^2 );
    end

    Rj=distancias_medidas(1);

    posicion=(S'*S)\(S'*(u-Rj*r));
    posicion_est(1)=abs(posicion(1));
    posicion_est(2)=abs(posicion(2));
    posicion_est(3)=abs(posicion(3));

    end

```

ESTIMACIÓN LEVENBERG

function posicion =estimacion_levenberg(distancias_medidas, puntos_medidos)

```

tiempos_medidos=distancias_medidas/3e8;

```

```

if length(tiempos_medidos)==6

```

```

    dif21=tiempos_medidos(2)-tiempos_medidos(1);
    dif31=tiempos_medidos(3)-tiempos_medidos(1);
    dif41=tiempos_medidos(4)-tiempos_medidos(1);
    dif51=tiempos_medidos(5)-tiempos_medidos(1);

```

```

    dif61=tiempos_medidos(6)-tiempos_medidos(1);

```

```

    x0=[0 0 0];

```

```

    f = @(x)funciones_6datos(x,puntos_medidos,dif21,dif31,dif41,dif51,dif61);
    options=optimset('Algorithm','levenberg-marquardt');
    posicion_estimada= fsolve(f,x0,options);
    posicion(1)=abs(posicion_estimada(1));

```

```

posicion(2)=abs(posicion_estimada(2));
posicion(3)=abs(posicion_estimada(3));

elseif length(tiempos_medidos)==5

    dif21=tiempos_medidos(2)-tiempos_medidos(1);
    dif31=tiempos_medidos(3)-tiempos_medidos(1);
    dif41=tiempos_medidos(4)-tiempos_medidos(1);
    dif51=tiempos_medidos(5)-tiempos_medidos(1);

    x0=[0 0 0];

    f = @(x)funciones_5datos(x,puntos_medidos,dif21,dif31,dif41,dif51);

    options=optimset('Algorithm','levenberg-marquardt');
    posicion_estimada= fsolve(f,x0,options);
    posicion(1)=abs(posicion_estimada(1));
    posicion(2)=abs(posicion_estimada(2));
    posicion(3)=abs(posicion_estimada(3));

elseif length(tiempos_medidos)==4

    dif21=tiempos_medidos(2)-tiempos_medidos(1);
    dif31=tiempos_medidos(3)-tiempos_medidos(1);
    dif41=tiempos_medidos(4)-tiempos_medidos(1);

    x0=[0 0 0];

    f = @(x)funciones_4datos(x,puntos_medidos,dif21,dif31,dif41);
    options=optimset('Algorithm','levenberg-marquardt');
    [posicion_estimada]= fsolve(f,x0,options);
    posicion(1)=abs(posicion_estimada(1));
    posicion(2)=abs(posicion_estimada(2));
    posicion(3)=abs(posicion_estimada(3));

else

    print('Casuística no definida')

end

```

ALGORITMO DE TAYLOR

```

function [posicion] =
TAYLOR(Puntos,TimeDOA,estimationposition,maxiterations,maxerror)

c = 299792458;

```

```
Q = cov(TimeDOA);
```

```
%Primero se muestra el caso de interés: 3D
```

```
if length(Puntos (1,:)) == 3
```

```
    if estimationposition == [0,0,0]
```

```
        x0 = mean(Puntos (:,1));
```

```
        y0 = mean(Puntos (:,2));
```

```
        z0 = mean(Puntos (:,3));
```

```
    else
```

```
        x0 = estimationposition(1);
```

```
        y0 = estimationposition(2);
```

```
        z0 = estimationposition(3);
```

```
    end
```

```
    for i=1:length(TimeDOA(:,1))
```

```
        for n=1:length(TimeDOA(1,:))
```

```
            tx1(n) = TimeDOA(i,n);
```

```
            rx1(n) = c*tx1(n);
```

```
        end
```

```
        j=0;
```

```
        y = 1;
```

```
        while (j<maxiterations && y > maxerror)
```

```
            for n=1:length(Puntos (:,1))
```

```
                r(n) = sqrt( (Puntos (n,1)-x0)^2 + (Puntos (n,2)-y0)^2 +
(Puntos (n,3)-z0)^2 );
```

```
            end
```

```
            for n=1:length(TimeDOA(1,:))
```

```
                ht(n,:) = [rx1(n) - (r(n+1)-r(1))];
```

```
                Gt(n,:) = [(Puntos (1,1)-x0)/r(1) - (Puntos (n+1,1)-
x0)/r(n+1), (Puntos (1,2)-y0)/r(1) - (Puntos (n+1,2)-y0)/r(n+1), (Puntos (1,3)-z0)/r(1) -
(Puntos (n+1,3)-z0)/r(n+1)];
```

```
            end
```

```
        A = inv(transpose(Gt)*inv(Q)*Gt)*transpose(Gt)*inv(Q)*ht;
```

```

        y = sqrt(A(1)^2 + A(2)^2 + A(3)^2);

        x0 = x0 + A(1);
        y0 = y0 + A(2);
        z0 = z0 + A(3);

        j = j+1;
    end

    posicion(i,:) = [abs(x0), abs(y0), abs(z0)];

    if estimationposition == [0,0,0]

        x0 = mean(Puntos(:,1));
        y0 = mean(Puntos(:,2));
        z0 = mean(Puntos(:,3));

    else

        x0 = estimationposition(1);
        y0 = estimationposition(2);
        z0 = estimationposition(3);

    end
end
end

```

%Segundo se muestra el caso 2D

```

elseif length(Puntos(1,:)) == 2

    if estimationposition == [0,0]

        x0 = mean(Puntos(:,1));
        y0 = mean(Puntos(:,2));

    else

        x0 = estimationposition(1);
        y0 = estimationposition(2);

    end

    for i=1:length(TimeDOA(:,1))

        for n=1:length(TimeDOA(1,:))

            tx1(n) = TimeDOA(i,n);
            rx1(n) = c*tx1(n);

        end

        j=0;
    end

```

```

y = 1;

while (j<maxiterations & y > maxerror)

    for n=1:length(Puntos(:,1))

        r(n) = sqrt( (Puntos(n,1)-x0)^2 + (Puntos(n,2)-y0)^2);

    end

    for n=1:length(TimeDOA(1,:))

        ht(n,:) = [rx1(n) - (r(n+1)-r(1))];

        Gt(n,:) = [(Puntos(1,1)-x0)/r(1) - (Puntos(n+1,1)-
x0)/r(n+1), (Puntos(1,2)-y0)/r(1) - (Puntos(n+1,2)-y0)/r(n+1)];
    end

    A = inv(transpose(Gt)*inv(Q)*Gt)*transpose(Gt)*inv(Q)*ht;

    y = sqrt(A(1)^2 + A(2)^2);
    x0 = x0 + A(1);
    y0 = y0 + A(2);

    j = j+1;
end

posicion(i,:) = [abs(x0),abs(y0)];

if estimationposition == [0,0]

    x0 = mean(Puntos(:,1));
    y0 = mean(Puntos(:,2));

else

    x0 = estimationposition(1);
    y0 = estimationposition(2);

end
end
end
end

```

ALGORITMO DE STEEPEST DESCENT

```

function [posicion,descent,grafica_datos,grafica_theta] = descent
(distancias,puntos,inicial,max)

```

```

%Primero defino conceptos básicos

```

```

N=length(distancias);

grafica_datos=[];
grafica_theta=[];

for repeticion=1:max

    %Segundo tengo que definir los phis para poder calcular sus
    %derivadas

    syms x y z

    sinderivar=zeros(1,N-1);
    derivadas=zeros(N-1,3);

    for k=1:N-1

        %Calculo los phis y sus derivadas

        phi=(sqrt((x-puntos(k+1,1))^2 + (y-puntos(k+1,2))^2 +(z-puntos(k+1,3))^2 )-...
            sqrt((x-puntos(1,1))^2 + (y-puntos(1,2))^2 +(z-puntos(1,3))^2)) -...
            (distancias(k+1)-distancias(1)));

        sinderivar(1,k)=subs(phi,[x,y,z],[inicial(1),inicial(2),inicial(3)]);

        derivadax=diff(phi,x,1);
        derivaday=diff(phi,y,1);
        derivadaz=diff(phi,z,1);

        a=subs(derivadax,[x,y,z],[inicial(1),inicial(2),inicial(3)]);
        b=subs(derivaday,[x,y,z],[inicial(1),inicial(2),inicial(3)]);
        c=subs(derivadaz,[x,y,z],[inicial(1),inicial(2),inicial(3)]);

        derivadas(k,:)=[a,b,c];

    end

    %Ahora tengo que calcular los gradientes

    suma1=0;
    suma2=0;
    suma3=0;

    for i=1:N-1

        suma1=suma1 + (derivadas(i,1)*sinderivar(i));
        suma2=suma2 + (derivadas(i,2)*sinderivar(i));
        suma3=suma3 + (derivadas(i,3)*sinderivar(i));

```



```

end

g10=2*suma1;
g20=2*suma2;
g30=2*suma3;

%El tercer paso es calcular landa

arriba=0;
abajo=0;

for i=1:N-1

    arriba=arriba + ( sinderivar(i) * (g10*derivadas(i,1) + g20*derivadas(i,2) +
g30*derivadas(i,3)) );
    abajo=abajo + ((g10*derivadas(i,1) + g20*derivadas(i,2) + g30*derivadas(i,3))^2);

end

landa=arriba/abajo;

%Ahora calculo x1,y1,z1

x1=inicial(1)-landa*g10;
y1=inicial(2)-landa*g20;
z1=inicial(3)-landa*g30;

%Compruebo el valor de theta

for k=1:N-1;

    %calcula otra vez los phi

    phi=(sqrt((x-puntos(k+1,1))^2 + (y-puntos(k+1,2))^2 +(z-puntos(k+1,3))^2
)-...
    sqrt((x-puntos(1,1))^2 + (y-puntos(1,2))^2 +(z-puntos(1,3))^2)) -...
    (distancias(k+1)-distancias(1));

    valor=subs(phi,[x,y,z],[x1,y1,z1]);

    theta=(valor^2);

end
inicial=[x1,y1,z1];
if theta<=4e-3
    break
end

grafica_datos(repeticion,:)=inicial;
grafica_theta(repeticion,1)=theta;

```

end

```
punto=zeros(1,3);
punto(1)=abs(inicial(1));
punto(2)=abs(inicial(2));
punto(3)=abs(inicial(3));
descent=punto;
```

end

ALGORITMO DE INTERPOLACIÓN ESFÉRICA

function [posicion] = interpolacion_esferica(puntos,distancias)

```
N=size(puntos,1);
RL=puntos;
aux=distancias;
```

```
w=ones(N-1,1);
```

%primero los d que son diferencia de distancias TDOA

```
d=zeros(N-1,1);
for i=1:N-1
```

```
    d(i,1)=aux(i+1)-aux(1);
end
```

%ahora los R

```
R=zeros(N-1,1);
for i=2:N
```

```
    R(i-1,1)=sqrt( (RL(i,1)-RL(1,1))^2 + (RL(i,2)-RL(1,2))^2 + (RL(i,3)-
    RL(1,3))^2);
```

```
end
```

```
S=RL(2:N,:);
```

```
S(:,1)=S(:,1)-RL(1,1);
S(:,2)=S(:,2)-RL(1,2);
S(:,3)=S(:,3)-RL(1,3);
```

```
delta=R.^2-d.^2;
```

```

W=diag(w);

Ps=S*inv(S'*W*S)*S'*W;
Pso=eye(N-1)-Ps;

Rshat=(d'*Pso*W*Pso*delta)/(2*d'*Pso*W*Pso*d);

estimate=.5*inv(S'*W*S)*S'*W*(delta-2*Rshat*d);

posicion=estimate'+RL(1,:);

end

```

ALGORITMO KNN

Están incluidos los mapas de potencia que se han obtenido para el escenario de Jerónimo de Ayaz I explicados en el apartado 3.3.2.2. Primero se muestra el mapa de 4 antenas y posteriormente el de 5 antenas.

Solo se muestra el caso de $k=1$ ya que ha sido el único caso utilizado.

function [posicion_RSS,distancias,distancias_ord] = RSS(potencias_medidas,N)

if N==4

```

mapa=[1 1 1 -21.0635 -19.4741 -29.2234 -32.9616;
      2 1 1 -27.5426 -23.8162 -23.0466 -69.9176;
      3 1 1 -25.3705 -33.9006 -36.6804 -47.2689;
      4 1 1 -27.7484 -42.4568 -28.1233 -37.08;
      5 1 1 -25.2525 -32.1143 -25.4985 -35.5928;
      6 1 1 -22.9885 -38.4851 -35.0901 -49.4107;
      8 1 1 -38.6563 -42.6415 -29.4 -39.9288;
      9 1 1 -40.3217 -26.6292 -21.9128 -30.2228;
      10 1 1 -24.2262 -50.2146 -16.7569 -21.0382;
      11 1 1 -46.8613 -37.5562 -8.8929 -19.9826;
      12 1 1 -33.6921 -34.3595 -25.6717 -19.9902;
      1 2 1 -20.2954 -19.3511 -31.1102 -61.8854;
      2 2 1 -25.9774 -19.3751 -34.3037 -33.1349;
      3 2 1 -32.0837 -30.3763 -23.1366 -45.4108;
      4 2 1 -22.7739 -22.3063 -31.085 -43.1654;
      5 2 1 -25.6894 -34.4413 -29.6937 -39.5096;
      6 2 1 -43.3023 -25.9212 -32.0037 -34.1767;
      8 2 1 -25.9193 -30.0035 -33.1082 -28.3864;
      9 2 1 -48.0731 -28.5971 -26.9806 -28.6582;
      10 2 1 -26.2201 -45.3229 -21.9403 -23.567;
      11 2 1 -47.7838 -43.1647 -16.1752 -31.3686;
      12 2 1 -35.6625 -54.6719 -15.5191 -13.2417;
      1 3 1 -21.4604 -27.9436 -49.9357 -200;

```

2 3 1 -24.1887 -15.497 -31.5707 -32.1969;
 3 3 1 -20.2047 -27.5777 -32.46 -63.1244;
 4 3 1 -24.4187 -20.3689 -34.1283 -35.8641;
 5 3 1 -24.2553 -24.8177 -33.3855 -41.2537;
 6 3 1 -26.7178 -24.6326 -28.8345 -43.488;
 8 3 1 -31.1032 -44.1407 -23.8078 -26.4117;
 9 3 1 -40.7502 -26.0308 -25.3283 -38.5641;
 10 3 1 -29.624 -53.9297 -20.2037 -17.4428;
 11 3 1 -48.7037 -82.2463 -23.0719 -14.3064;
 12 3 1 -31.4033 -29.9326 -18.5564 -9.9391;
 2 4 1 -26.9998 -19.5182 -26.5606 -33.5212;
 3 4 1 -23.4756 -25.3403 -28.1816 -81.3434;
 4 4 1 -27.791 -23.574 -25.7687 -34.2864;
 5 4 1 -35.5072 -24.7059 -21.3511 -46.6468;
 6 4 1 -31.63 -26.4738 -25.4836 -30.9612;
 7 4 1 -37.1205 -35.9219 -47.6218 -25.6443;
 8 4 1 -34.0714 -30.0792 -26.8886 -32.4035;
 10 4 1 -36.7057 -37.0691 -26.718 -17.7367;
 11 4 1 -43.2291 -34.4569 -27.0584 -17.5886;
 12 4 1 -32.1481 -60.0648 -23.5727 -25.7224;
 1 1 2 -15.8529 -30.4989 -29.5885 -66.6471;
 2 1 2 -34.2827 -20.3665 -30.1364 -76.5879;
 3 1 2 -22.7974 -29.1274 -30.8978 -43.9007;
 4 1 2 -24.8606 -34.8943 -27.3803 -40.9816;
 5 1 2 -20.476 -24.0531 -26.3246 -52.7887;
 7 1 2 -33.1607 -32.843 -26.5301 -28.2767;
 8 1 2 -31.0283 -37.5779 -18.6855 -31.2025;
 9 1 2 -37.4502 -28.9407 -19.1872 -19.3354;
 10 1 2 -27.8722 -37.729 -9.6211 -24.4236;
 12 1 2 -32.843 -32.8621 -8.2462 -20.5068;
 1 2 2 -19.221 -18.5548 -27.3067 -60.0688;
 2 2 2 -16.1646 -21.5764 -34.2872 -33.622;
 3 2 2 -16.3926 -24.5074 -33.7967 -42.8716;
 4 2 2 -26.0908 -19.7027 -23.746 -44.1514;
 5 2 2 -22.1012 -28.5856 -20.2176 -40.0767;
 7 2 2 -33.7733 -26.8818 -27.2204 -51.2573;
 8 2 2 -30.3403 -36.9399 -25.6702 -28.2782;
 9 2 2 -58.1726 -31.4017 -23.644 -37.5904;
 10 2 2 -29.7564 -29.9206 -14.7193 -24.9142;
 11 2 2 -35.8845 -44.3853 -11.6596 -20.6963;
 12 2 2 -33.279 -36.1183 -19.4776 -19.1629;
 1 3 2 -28.4442 -14.2614 -41.1633 -200;
 2 3 2 -21.7079 -36.4823 -34.0498 -32.8045;
 3 3 2 -26.0998 -28.5286 -48.116 -56.1424;
 4 3 2 -18.6673 -18.6016 -35.3166 -31.006;
 5 3 2 -26.5908 -27.2558 -40.9365 -37.3322;
 7 3 2 -31.2369 -34.4252 -41.6422 -28.8075;
 8 3 2 -29.9844 -51.0223 -17.3638 -43.436;
 9 3 2 -37.6652 -47.3085 -18.6794 -29.485;
 10 3 2 -27.593 -56.8148 -16.2898 -22.3508;

11 3 2 -34.9685 -66.8964 -19.2509 -14.9337;
 12 3 2 -34.5439 -27.1006 -30.0328 -23.617;
 1 4 2 -23.7271 -9.2044 -27.1117 -31.8098;
 2 4 2 -18.7054 -18.5316 -34.0962 -38.4601;
 3 4 2 -21.5598 -22.0109 -30.77 -43.3932;
 4 4 2 -35.64 -21.4135 -38.5192 -34.4414;
 5 4 2 -35.4464 -22.1505 -26.1002 -43.087;
 6 4 2 -33.8915 -22.3276 -33.4939 -23.3639;
 7 4 2 -27.0911 -50.6462 -24.3448 -24.9941;
 9 4 2 -27.5975 -25.5523 -22.8116 -26.874;
 10 4 2 -42.8327 -33.0874 -34.6005 -22.9769;
 11 4 2 -61.4695 -47.0048 -24.3007 -18.0289;
 12 4 2 -28.6143 -136.4313 -35.0212 -10.4218;
 1 1 3 -3.4225 -18.1856 -30.9293 -33.7616 ;
 2 1 3 -5.7199 -32.0024 -35.4421 -71.3468 ;
 3 1 3 -14.5172 -35.224 -40.1809 -51.5842 ;
 4 1 3 -25.3609 -47.0817 -37.1248 -52.4799;
 5 1 3 -17.3454 -32.8295 -22.5774 -46.2118;
 6 1 3 -28.7975 -45.8117 -26.47 -37.2962 ;
 7 1 3 -37.2634 -39.1351 -22.081 -35.5747 ;
 8 1 3 -31.8001 -26.5779 -19.5772 -29.3098;
 9 1 3 -27.9607 -26.9564 -15.983 -32.4758 ;
 10 1 3 -27.3132 -58.5268 -16.4843 -20.1362;
 11 1 3 -42.4868 -33.5372 -4.3735 -20.2683 ;
 12 1 3 -40.8479 -39.0133 11.9864 -37.9752;
 1 2 3 -16.0095 -21.6406 -31.2936 -69.5415;
 2 2 3 -20.0613 -17.0921 -50.2426 -39.3725;
 3 2 3 -12.8485 -21.991 -34.425 -49.5563 ;
 4 2 3 -23.0388 -24.3077 -29.2716 -44.3105 ;
 5 2 3 -41.5076 -20.0115 -20.6235 -31.0785 ;
 6 2 3 -31.7992 -31.6899 -30.22 -32.8677 ;
 7 2 3 -41.3918 -26.3932 -21.9396 -33.2729 ;
 8 2 3 -59.308 -38.7767 -29.3572 -28.5261 ;
 9 2 3 -41.554 -30.5128 -29.2254 -42.5628 ;
 10 2 3 -28.7059 -32.1004 -13.3782 -22.0326;
 11 2 3 -52.1834 -32.6667 -7.9544 -23.9151;
 12 2 3 -32.8235 -32.9141 -8.9172 -21.1751;
 1 3 3 -14.131 -13.0278 -53.5117 -86.4986 ;
 2 3 3 -21.0171 -30.5211 -43.072 -58.8358 ;
 3 3 3 -16.0328 -26.5506 -35.6173 -41.6964 ;
 4 3 3 -20.0983 -30.7749 -34.1675 -30.4749 ;
 6 3 3 -22.937 -24.5726 -19.3115 -43.4619;
 7 3 3 -28.5888 -47.5045 -30.6899 -35.6557;
 8 3 3 -30.333 -28.4392 -22.5582 -35.0589;
 9 3 3 -26.8948 -29.8874 -16.4898 -28.3743;
 10 3 3 -26.6089 -45.1628 -12.1608 -22.427;
 11 3 3 -33.444 -43.87 -23.9917 -13.1686;
 12 3 3 -43.8293 -30.1018 -29.5056 -7.1066;
 1 4 3 -19.9519 -4.6397 -43.0526 -200;
 2 4 3 -16.4289 -25.0008 -29.5821 -61.8267;

```

3 4 3 -20.8928    -14.3367 -29.2482 -73.8356;
4 4 3 -23.8709    -33.5982 -43.8135 -37.1087;
5 4 3 -23.2035 -23.6904 -26.3305 -42.226;
6 4 3 -39.1993 -24.5799 -30.5858 -32.44;
7 4 3 -32.5038 -30.9687 -18.5369 -25.7405;
8 4 3 -23.8046 -29.4921 -16.2686 -30.6379;
9 4 3 -25.9264 -28.748    -27.843    -36.9949;
10 4 3 -51.3125 -35.596    -16.1156 -26.239;
11 4 3 -52.8449 -39.3078 -22.2771 -20.3521];

```

```
elseif N==5
```

```

mapa=[1 1 1 -21.0635 -19.4741 -29.2234 -32.9616 -32.3354;
2 1 1 -27.5426 -23.8162 -23.0466 -69.9176 -26.0412;
3 1 1 -25.3705 -33.9006 -36.6804 -47.2689 -17.749;
4 1 1 -27.7484 -42.4568 -28.1233 -37.08 -19.1507;
5 1 1 -25.2525 -32.1143 -25.4985 -35.5928 -19.6767;
6 1 1 -22.9885 -38.4851 -35.0901 -49.4107 -18.2092;
8 1 1 -38.6563 -42.6415 -29.4    -39.9288 -25.9724;
9 1 1 -40.3217 -26.6292 -21.9128 -30.2228 -20.6271;
10 1 1 -24.2262 -50.2146 -16.7569 -21.0382 -20.8393;
11 1 1 -46.8613 -37.5562 -8.8929 -19.9826 -22.9465;
12 1 1 -33.6921 -34.3595 -25.6717 -19.9902 -22.2501;
1 2 1 -20.2954 -19.3511 -31.1102 -61.8854 -20.3674;
2 2 1 -25.9774 -19.3751 -34.3037 -33.1349 -21.8;
3 2 1 -32.0837 -30.3763 -23.1366 -45.4108 -18.8143;
4 2 1 -22.7739 -22.3063 -31.085 -43.1654 -19.5255;
5 2 1 -25.6894 -34.4413 -29.6937 -39.5096 -28.7969;
6 2 1 -43.3023 -25.9212 -32.0037 -34.1767 -21.6459;
8 2 1 -25.9193 -30.0035 -33.1082 -28.3864 -17.6429;
9 2 1 -48.0731 -28.5971 -26.9806 -28.6582 -21.2604;
10 2 1 -26.2201 -45.3229 -21.9403 -23.567 -48.7779;
11 2 1 -47.7838 -43.1647 -16.1752 -31.3686 -20.479;
12 2 1 -35.6625 -54.6719 -15.5191 -13.2417 -20.2535;
1 3 1 -21.4604 -27.9436 -49.9357 -200 -32.3513;
2 3 1 -24.1887 -15.497 -31.5707 -32.1969 -21.9682;
3 3 1 -20.2047 -27.5777 -32.46 -63.1244 -25.4065;
4 3 1 -24.4187 -20.3689 -34.1283 -35.8641 -17.61;
5 3 1 -24.2553 -24.8177 -33.3855 -41.2537 -22.4366;
6 3 1 -26.7178 -24.6326 -28.8345 -43.488 -17.9175;
8 3 1 -31.1032 -44.1407 -23.8078 -26.4117 -30.1699;
9 3 1 -40.7502 -26.0308 -25.3283 -38.5641 -27.9597;
10 3 1 -29.624    -53.9297 -20.2037 -17.4428 -23.3881;
11 3 1 -48.7037 -82.2463 -23.0719 -14.3064 -37.3004;
12 3 1 -31.4033 -29.9326 -18.5564 -9.9391 -30.463;
2 4 1 -26.9998 -19.5182 -26.5606 -33.5212 -47.3775;
3 4 1 -23.4756 -25.3403 -28.1816 -81.3434 -23.1324;
4 4 1 -27.791    -23.574    -25.7687 -34.2864 -21.5232;
5 4 1 -35.5072 -24.7059 -21.3511 -46.6468 -31.3137;
6 4 1 -31.63 -26.4738    -25.4836 -30.9612 -16.4752;

```

7 4 1 -37.1205 -35.9219 -47.6218 -25.6443 -29.3357;
 8 4 1 -34.0714 -30.0792 -26.8886 -32.4035 -22.4523;
 10 4 1 -36.7057 -37.0691 -26.718 -17.7367 -34.0549;
 11 4 1 -43.2291 -34.4569 -27.0584 -17.5886 -29.519;
 12 4 1 -32.1481 -60.0648 -23.5727 -25.7224 -29.578;
 1 1 2 -15.8529 -30.4989 -29.5885 -66.6471 -26.4825;
 2 1 2 -34.2827 -20.3665 -30.1364 -76.5879 -20.8188;
 3 1 2 -22.7974 -29.1274 -30.8978 -43.9007 -19.6375;
 4 1 2 -24.8606 -34.8943 -27.3803 -40.9816 -14.6446;
 5 1 2 -20.476 -24.0531 -26.3246 -52.7887 -12.2176;
 7 1 2 -33.1607 -32.843 -26.5301 -28.2767 -40.9837;
 8 1 2 -31.0283 -37.5779 -18.6855 -31.2025 -21.0427;
 9 1 2 -37.4502 -28.9407 -19.1872 -19.3354 -17.3134;
 10 1 2 -27.8722 -37.729 -9.6211 -24.4236 -19.0764;
 12 1 2 -32.843 -32.8621 -8.2462 -20.5068 -21.5227;
 1 2 2 -19.221 -18.5548 -27.3067 -60.0688 -26.358;
 2 2 2 -16.1646 -21.5764 -34.2872 -33.622 -34.9535;
 3 2 2 -16.3926 -24.5074 -33.7967 -42.8716 -19.9377;
 4 2 2 -26.0908 -19.7027 -23.746 -44.1514 -15.3028;
 5 2 2 -22.1012 -28.5856 -20.2176 -40.0767 -16.151;
 7 2 2 -33.7733 -26.8818 -27.2204 -51.2573 -25.0618;
 8 2 2 -30.3403 -36.9399 -25.6702 -28.2782 -23.6176;
 9 2 2 -58.1726 -31.4017 -23.644 -37.5904 -25.2782;
 10 2 2 -29.7564 -29.9206 -14.7193 -24.9142 -23.8913;
 11 2 2 -35.8845 -44.3853 -11.6596 -20.6963 -25.3998;
 12 2 2 -33.279 -36.1183 -19.4776 -19.1629 -20.3238;
 1 3 2 -28.4442 -14.2614 -41.1633 -200 -26.6488;
 2 3 2 -21.7079 -36.4823 -34.0498 -32.8045 -38.4223;
 3 3 2 -26.0998 -28.5286 -48.116 -56.1424 -28.5394;
 4 3 2 -18.6673 -18.6016 -35.3166 -31.006 -22.6533;
 5 3 2 -26.5908 -27.2558 -40.9365 -37.3322 -20.2602;
 7 3 2 -31.2369 -34.4252 -41.6422 -28.8075 -16.9202;
 8 3 2 -29.9844 -51.0223 -17.3638 -43.436 -23.26;
 9 3 2 -37.6652 -47.3085 -18.6794 -29.485 -15.9479;
 10 3 2 -27.593 -56.8148 -16.2898 -22.3508 -17;
 11 3 2 -34.9685 -66.8964 -19.2509 -14.9337 -48.0283;
 12 3 2 -34.5439 -27.1006 -30.0328 -23.617 -21.1198;
 1 4 2 -23.7271 -9.2044 -27.1117 -31.8098 -29.0083;
 2 4 2 -18.7054 -18.5316 -34.0962 -38.4601 -23.9046;
 3 4 2 -21.5598 -22.0109 -30.77 -43.3932 -25.9026;
 4 4 2 -35.64 -21.4135 -38.5192 -34.4414 -17.9139;
 5 4 2 -35.4464 -22.1505 -26.1002 -43.087 -16.0033;
 6 4 2 -33.8915 -22.3276 -33.4939 -23.3639 -24.6967;
 7 4 2 -27.0911 -50.6462 -24.3448 -24.9941 -17.7594;
 9 4 2 -27.5975 -25.5523 -22.8116 -26.874 -19.9462;
 10 4 2 -42.8327 -33.0874 -34.6005 -22.9769 -34.5013;
 11 4 2 -61.4695 -47.0048 -24.3007 -18.0289 -37.1434;
 12 4 2 -28.6143 -136.4313 -35.0212 -10.4218 -31.75;
 1 1 3 -3.4225 -18.1856 -30.9293 -33.7616 -25.4339;
 2 1 3 -5.7199 -32.0024 -35.4421 -71.3468 -29.1686;

```

3 1 3 -14.5172    -35.224      -40.1809 -51.5842 -18.0676;
4 1 3 -25.3609 -47.0817 -37.1248 -52.4799 -15.4633;
5 1 3 -17.3454    -32.8295 -22.5774 -46.2118 -23.5742;
6 1 3 -28.7975 -45.8117 -26.47 -37.2962 -9.1816;
7 1 3 -37.2634    -39.1351 -22.081 -35.5747 -13.9764;
8 1 3 -31.8001 -26.5779 -19.5772 -29.3098 -23.0404;
9 1 3 -27.9607    -26.9564 -15.983 -32.4758 -18.1268;
10 1 3 -27.3132 -58.5268 -16.4843 -20.1362 -24.323;
11 1 3 -42.4868    -33.5372 -4.3735 -20.2683 -31.1061;
12 1 3 -40.8479 -39.0133 11.9864 -37.9752 -21.5901;
1 2 3 -16.0095    -21.6406 -31.2936 -69.5415 -20.3485;
2 2 3 -20.0613 -17.0921 -50.2426 -39.3725 -26.7527;
3 2 3 -12.8485 -21.991    -34.425      -49.5563 -17.4306;
4 2 3 -23.0388 -24.3077 -29.2716 -44.3105 -22.1989;
5 2 3 -41.5076    -20.0115 -20.6235 -31.0785 -18.0907;
6 2 3 -31.7992 -31.6899 -30.22 -32.8677 1.1144;
7 2 3 -41.3918    -26.3932 -21.9396 -33.2729 -15.7295;
8 2 3 -59.308 -38.7767 -29.3572 -28.5261 -19.8766;
9 2 3 -41.554 -30.5128    -29.2254 -42.5628 -18.2674;
10 2 3 -28.7059 -32.1004 -13.3782 -22.0326 -18.4529;
11 2 3 -52.1834    -32.6667 -7.9544 -23.9151 -23.8029;
12 2 3 -32.8235 -32.9141 -8.9172 -21.1751 -27.8547;
    1 3 3 -14.131 -13.0278    -53.5117 -86.4986 -27.3049;
2 3 3 -21.0171    -30.5211 -43.072 -58.8358 -19.621;
3 3 3 -16.0328 -26.5506 -35.6173 -41.6964 -18.0448;
4 3 3 -20.0983    -30.7749 -34.1675 -30.4749 -21.0034;
6 3 3 -22.937 -24.5726    -19.3115 -43.4619 -11.9778;
7 3 3 -28.5888 -47.5045 -30.6899 -35.6557 -9.4709;
8 3 3 -30.333 -28.4392    -22.5582 -35.0589 -16.579;
9 3 3 -26.8948 -29.8874 -16.4898 -28.3743 -20.1174;
10 3 3 -26.6089    -45.1628 -12.1608 -22.427 -22.3894;
11 3 3 -33.444 -43.87 -23.9917 -13.1686 -44.2226;
12 3 3 -43.8293 -30.1018 -29.5056 -7.1066 -29.833;
1 4 3 -19.9519 -4.6397 -43.0526 -200 -19.0117;
2 4 3 -16.4289    -25.0008 -29.5821 -61.8267 -24.2522;
3 4 3 -20.8928    -14.3367 -29.2482 -73.8356 -23.7126;
4 4 3 -23.8709    -33.5982 -43.8135 -37.1087 -9.5752;
5 4 3 -23.2035 -23.6904 -26.3305 -42.226 -25.0606;
6 4 3 -39.1993 -24.5799 -30.5858 -32.44 -8.0736;
7 4 3 -32.5038 -30.9687 -18.5369 -25.7405 -15.3661;
8 4 3 -23.8046 -29.4921 -16.2686 -30.6379 -20.3734;
9 4 3 -25.9264 -28.748    -27.843      -36.9949 -20.0524;
10 4 3 -51.3125 -35.596    -16.1156 -26.239 -19.0601;
11 4 3 -52.8449 -39.3078 -22.2771 -20.3521 -21.1794];

```

end

%Tengo como variables el mapa que es un conjunto de filas con posicion y las
potencias recibidas por
%los sensores.


```

%Por otro lado tenemos las potencias_medidas para poder hallar el punto de
%emision
numero_sensores=length(potencias_medidas);

%Una vez que tenemos el mapa de potencia y las potencias terminal
%tenemos que ejecutar el algoritmo de k nearest neighbors
%Empezamos calculando las distancias entre las potencias calculadas y
%las del terminal
[a,b]=size(mapa);

distancias=zeros(1,a);
for i=1:a %para cada punto del mapa calculo distancia al punto a buscar
    recuento=0;
    for j=1:numero_sensores
        recuento=recuento + (mapa(i,j+3)-potencias_medidas(j))^2;
    end
    distancias(i)=sqrt(recuento);
end

%Con el vector de distancias calculado ya puedo buscar el minimo que
%será la mejor aproximacion al terminal

[valor,posicion]=min(distancias);
distancias_ord=sort(distancias,2);
[posicion_RSS]=mapa(posicion,1:3);

end

```

PROGRAMAS COMPLEMENTARIOS A LOS ALGORITMOS ANTERIORES

```

function f= funciones_4datos(x,puntos_medidos,dif21,dif31,dif41)

c=3e8;

f=[1/c*(sqrt((x(1)-puntos_medidos(2,1))^2 + (x(2)-puntos_medidos(2,2))^2 + (x(3)-
puntos_medidos(2,3))^2)-...

    sqrt((x(1)-puntos_medidos(1,1))^2 + (x(2)-puntos_medidos(1,2))^2 + (x(3)-
puntos_medidos(1,3))^2)) - dif21;

    1/c*(sqrt((x(1)-puntos_medidos(3,1))^2 + (x(2)-puntos_medidos(3,2))^2 + (x(3)-
puntos_medidos(3,3))^2)-...

    sqrt((x(1)-puntos_medidos(1,1))^2 + (x(2)-puntos_medidos(1,2))^2 + (x(3)-
puntos_medidos(1,3))^2)) - dif31;

```

$1/c * (\sqrt{(x(1) - \text{puntos_medidos}(4,1))^2 + (x(2) - \text{puntos_medidos}(4,2))^2 + (x(3) - \text{puntos_medidos}(4,3))^2}) - \dots$

$\sqrt{(x(1) - \text{puntos_medidos}(1,1))^2 + (x(2) - \text{puntos_medidos}(1,2))^2 + (x(3) - \text{puntos_medidos}(1,3))^2}) - \text{dif41}]$;

end

function f= funciones_5datos(x,puntos_medidos,dif21,dif31,dif41,dif51)

c=3e8;

f=[1/c*(sqrt((x(1)-puntos_medidos(2,1))^2 + (x(2)-puntos_medidos(2,2))^2 + (x(3)-puntos_medidos(2,3))^2)-...

sqrt((x(1)-puntos_medidos(1,1))^2 + (x(2)-puntos_medidos(1,2))^2 + (x(3)-puntos_medidos(1,3))^2)) - dif21;

1/c*(sqrt((x(1)-puntos_medidos(3,1))^2 + (x(2)-puntos_medidos(3,2))^2 + (x(3)-puntos_medidos(3,3))^2)-...

sqrt((x(1)-puntos_medidos(1,1))^2 + (x(2)-puntos_medidos(1,2))^2 + (x(3)-puntos_medidos(1,3))^2)) - dif31;

1/c*(sqrt((x(1)-puntos_medidos(4,1))^2 + (x(2)-puntos_medidos(4,2))^2 + (x(3)-puntos_medidos(4,3))^2)-...

sqrt((x(1)-puntos_medidos(1,1))^2 + (x(2)-puntos_medidos(1,2))^2 + (x(3)-puntos_medidos(1,3))^2)) - dif41;

1/c*(sqrt((x(1)-puntos_medidos(5,1))^2 + (x(2)-puntos_medidos(5,2))^2 + (x(3)-puntos_medidos(5,3))^2)-...

sqrt((x(1)-puntos_medidos(1,1))^2 + (x(2)-puntos_medidos(1,2))^2 + (x(3)-puntos_medidos(1,3))^2)) - dif51];

end

function f= funciones_6datos(x,puntos_medidos,dif21,dif31,dif41,dif51,dif61)

c=3e8;

f=[1/c*(sqrt((x(1)-puntos_medidos(2,1))^2 + (x(2)-puntos_medidos(2,2))^2 + (x(3)-puntos_medidos(2,3))^2)-...

sqrt((x(1)-puntos_medidos(1,1))^2 + (x(2)-puntos_medidos(1,2))^2 + (x(3)-puntos_medidos(1,3))^2)) - dif21;

$1/c * (\sqrt{(x(1)-\text{puntos_medidos}(3,1))^2 + (x(2)-\text{puntos_medidos}(3,2))^2 + (x(3)-\text{puntos_medidos}(3,3))^2}) - \dots$

$\sqrt{(x(1)-\text{puntos_medidos}(1,1))^2 + (x(2)-\text{puntos_medidos}(1,2))^2 + (x(3)-\text{puntos_medidos}(1,3))^2}) - \text{dif31};$

$1/c * (\sqrt{(x(1)-\text{puntos_medidos}(4,1))^2 + (x(2)-\text{puntos_medidos}(4,2))^2 + (x(3)-\text{puntos_medidos}(4,3))^2}) - \dots$

$\sqrt{(x(1)-\text{puntos_medidos}(1,1))^2 + (x(2)-\text{puntos_medidos}(1,2))^2 + (x(3)-\text{puntos_medidos}(1,3))^2}) - \text{dif41};$

$1/c * (\sqrt{(x(1)-\text{puntos_medidos}(5,1))^2 + (x(2)-\text{puntos_medidos}(5,2))^2 + (x(3)-\text{puntos_medidos}(5,3))^2}) - \dots$

$\sqrt{(x(1)-\text{puntos_medidos}(1,1))^2 + (x(2)-\text{puntos_medidos}(1,2))^2 + (x(3)-\text{puntos_medidos}(1,3))^2}) - \text{dif51};$

$1/c * (\sqrt{(x(1)-\text{puntos_medidos}(6,1))^2 + (x(2)-\text{puntos_medidos}(6,2))^2 + (x(3)-\text{puntos_medidos}(6,3))^2}) - \dots$

$\sqrt{(x(1)-\text{puntos_medidos}(1,1))^2 + (x(2)-\text{puntos_medidos}(1,2))^2 + (x(3)-\text{puntos_medidos}(1,3))^2}) - \text{dif61};$

end

function output=depurar_datos(puntos,input,moda)

X = max(puntos(:,1));

Y = max(puntos(:,2));

Z = max(puntos(:,3));

Filas_nok = 0;

for i=1:length(input(:,1))

if abs(input(i,1)) > X | abs(input(i,2)) > Y | abs(input(i,3)) > Z

Filas_nok (length(Filas_nok)+1) = i;

end

end

if length(Filas_nok) > 1

```
Filas_nok = Filas_nok (2:end);

input = Filas_nok (input, Filas_nok);

end

if moda == 1

    [nx,xout] = hist(input(:,1));

    [Cx,Ix] = max(nx);

    output(1) = xout(Ix(1));

    [ny,yout] = hist(input(:,2));

    [Cy,Iy] = max(ny);

    output(2) = yout(Iy(1));

    [nz,zout] = hist(input(:,3));

    [Cz,Iz] = max(nz);

    output(3) = zout(Iz(1));

else

    output(1) = mean(input(:,1));

    output(2) = mean(input(:,2));

    output(3) = mean(input(:,3));

end

end
```

